2020

A sample weight and adaboost CNN-based coarse to fine classification of fruit and vegetables at a supermarket self-checkout

Khurram Hameed
Edith Cowan University

Douglas Chai
Edith Cowan University

Alexander Rassau
Edith Cowan University

Article

A Sample Weight and AdaBoost CNN-Based Coarse to Fine Classification of Fruit and Vegetables at a Supermarket Self-Checkout

Khurram Hameed * , Douglas Chai and Alexander Rassau

School of Engineering, Edith Cowan University, 270 Joondalup Drive, Joondalup, Perth, WA 6027, Australia; d.chai@ecu.edu.au (D.C.); a.rassau@ecu.edu.au (A.R.)

* Correspondence: k.hameed@ecu.edu.au

check for updates

Abstract: The physical features of fruit and vegetables make the task of vision-based classification of fruit and vegetables challenging. The classification of fruit and vegetables at a supermarket self-checkout poses even more challenges due to variable lighting conditions and human factors arising from customer interactions with the system along with the challenges associated with the colour, texture, shape, and size of a fruit or vegetable. Considering this complex application, we have proposed a progressive coarse to fine classification technique to classify fruit and vegetables at supermarket checkouts. The image and weight of fruit and vegetables have been obtained using a prototype designed to simulate the supermarket environment, including the lighting conditions. The weight information is used to change the coarse classification of 15 classes down to three, which are further used in AdaBoost-based Convolutional Neural Network (CNN) optimisation for fine classification. The training samples for each coarse class are weighted based on AdaBoost optimisation, which are updated on each iteration of a training phase. Multi-class likelihood distribution obtained by the fine classification stage is used to estimate a final classification with a softmax classifier. GoogleNet, MobileNet, and a custom CNN have been used for AdaBoost optimisation, with promising classification results.

Keywords: adaboost cnn optimisation; image classification; convolutional neural networks; supermarket self-checkouts; fruit and vegetables classification; progressive classification

1. Introduction

Current supermarket self-checkouts depend upon barcode scanning or selection from a Look Up Table (LUT) for billing. Packaged products at supermarkets can easily support barcodes, however fruit and vegetables, i.e., fresh produce items, must currently be selected from a LUT either by the assisted checkout personnel or by the customer at a self-checkout. This selection from a LUT involves significant human factors and requires good knowledge of different fruit and vegetable varieties. Fruit and vegetables are among the most sold produce items and have a significant contribution in the revenue of supermarkets and, hence, the economy of a country. For example, Australian supermarkets are a AUD 101 billion industry according to the IBISWorld Senior Industry Analyst [1,2]. This industry is also an employer of approximately 360,000 personnel across the nation. Given the size of the industry, intentional or unintentional incorrect scanning of fruit and vegetables can cause significant losses that can aggregate across the sector. Hence, the introduction of an image-based technique, as proposed in this paper, that eliminate the requirement for a LUT, can significantly improve revenues. The proposed technique also has significant environmental benefits by reducing the use of light-weight plastic packaging and shrink warps, which are currently used to locate barcodes. This plastic waste is an

exponentially growing problem all over the world. For instance, approximately 3.5 million tonnes of plastic waste is produced in Australia annually and 0.6 million tonnes was produced via packaging in 2016–2017 [3]. Most of this plastic is not recycled, and as well as going into landfills, a significant percentage of this waste makes its way to sea. Recently, it has been estimated that there will be approximately 12 million kg of plastic waste in international oceans by 2050 [4]. The Environmental Protection Authority (EPA) of Australia recently reported that approximately 75% of low weight plastic is produced by plastic bags and packaging in supermarkets. Considering these factors, there is a strong justification to support the concept of a barcode-less supermarket self-checkout.

Fruit and vegetable classification is a complex problem and involves significant challenges. At a higher level of abstraction these challenges can be categorised as: (a) Classification of different fruit and vegetables and (b) classification of different varieties of a fruit or vegetable. The challenges for vision-based classification result from the highly variant physical features of fruit and vegetables i.e., level of ripeness, texture, colour, and shape. However, classification of fruit and vegetables at supermarket self-checkouts presents additional challenges such as variable ambient lighting conditions, human elements in the scanning process, and scanning of multiple fruit and vegetables at the same time. Much research has been published to discuss the design and implementation of automated supermarket self-checkouts [5–8]. However, a complete discussion on the classification of multiple fruit and vegetables in a supermarket environment is required to analyse the effectiveness of the concept. Moreover, the existing techniques have analysed the classification of fruit and vegetables by using vision-based information only. However, the weight of a fruit or vegetable is also available with the help of a built-in weight sensor at the supermarket checkout counter. This weight information has not previously been considered for classification purposes. Therefore, we propose a novel approach to incorporate the weight information of a fruit or vegetable for classification. A comparison of recent state-of-the-art features and Machine Learning (ML) techniques for fruit and vegetables, can be seen in Table 1 for our proposed approach. An implication can be observed in that much of the existing state-of-the-art work has been performed for small numbers of fruit and vegetable classes with small data sets, which can cause overfitting. In this paper, we propose a progressive fruit and vegetable classification technique for supermarket self-checkouts. Fruit and vegetable images are initially grouped based on the average weight of each fruit or vegetable class so as to give a coarse classification. These coarse classes are further processed with AdaBoost-based optimisation of Convolutional Neural Networks (CNNs) for fine classification.

Table 1. Comparison of recent state-of-the-art features and Machine Learning (ML) techniques for fruit and vegetable classification.

Ref.	Year	Fruit/Vegetable	Features	ML Technique	Accuracy (%)
[9]	2016	Rice Crop	Morphology, height, length	KNN	87.9
[10]	2017	FoodCast dataset	Colour mean and variance	Naive Bayes	73.0
[11]	2017	Radish	Spectral features	Discriminant Analysis	74.4
[12]	2017	Wheat	Texture approximation	Discriminant Analysis	77.0
[13]	2017	Grapes	Correlation similarity matrix	K-Means	86.8
[14]	2018	Cucumber	Blob centroid	Pixel SVM	85.6
[15]	2019	Date fruit	Deep texture feature	AlexNet	92.3
[16]	2020	Fruit and vegetables	HSV colour transforms	SVM	92.7
[17]	2020	Lettuce	Deep CNN features	DarkNet	93.0
This paper		Fruit and vegetables	Sample weight Deep CNN features	Jenks Natural Breaks AdaBoost Optimised CNN	93.9

The rest of the article is organised as follows. An overview of the state-of-the-art techniques of fruit and vegetables classification along with their applications is presented in Section 2. A prototype design to emulate the placement of fruit or vegetables at a self-checkout for billing with typical supermarket ambient lighting conditions is discussed in Section 3.1. A process of weight and image data acquisition and their organisation for further processing is explained in Section 3.2. A progressive

coarse to fine classification-based methodology for fruit and vegetables classification is discussed in Section 4. The implementation of the proposed technique and the experimental results are presented in Section 5. A detailed discussion on the results obtained and future applications of the proposed approach for real-world supermarket self-checkouts can be found in Section 6.

2. Literature Review

The vision-based classification of fruit and vegetables has been performed in many fields for a range of different applications. The most common applications include the classification of fruit or vegetables for automated harvesting in agricultural settings [18–20] or vision-based quality assessment of fruit or vegetables [21–23].

2.1. Robotic Harvesting

DarkNet has been used for the classification of lettuces for robotic harvesting in [17]. The lettuces were initially identified with a You Only Look Once (YOLO3) CNN, where the image of each identified lettuce is further processed for Representation Learning (RL) and classification. A classification accuracy of 82% is obtained for the harvesting and grading of lettuces. A pixel accumulation-based rice crop classification has been reported in [24]. A combination of two cameras was used for imaging and crop boundary estimation. Recently, multiple cameras were used to estimate the 3D coordinates of banana bunches in an orchard in [25]. A triangulation technique has been used for picking point estimation. A detailed review on vision-based fruit localisation and picking techniques can be found in [26,27]. The maturity of date fruit is estimated for making harvesting decisions in [15]. A multi-class classification frame work is defined based on transfer learned AlexNet and VGGNet [28]. The multi-class classification obtained from the Alex and VGG Nets then becomes an input of a binary classifier for making decision related to harvesting. A modified classifier block is used with VGGNet for the classification of date fruit in [29]. The date fruit was classified based on the maturity level and surface defects, where an accuracy of 96.98% was reported. A compression of statistical and CNN-based features is performed in [30] for recognition of food types. Two Support Vector Machine (SVM) classifiers were trained based on two kinds of features extracted by statistical techniques and CNN, where a respective accuracy of 93.03% and 94.01% were obtained.

2.2. Quality Grading

A colour-based citrus fruit quality assessment has been performed in [31], where three dominant colours of the obtained images are estimated by K-means clustering with different cluster sizes. RGB colour gradient, variance, and chromatic coordinates are used as features for correlation with standard quality parameters of citrus fruits. Statistical and Artificial Neural Network (ANN)-based techniques have been used to estimate Bayesian regulation, Levenberg Marquardt, and gradient descent as correlation parameters. A vision-based diseased Papaya fruit detection is performed in [32], where Grey Level Co-occurrence Matrix (GLCM)-based statistical features are extracted. These extracted features are classified with a SVM for diseased fruit identification. A ResNet-based classification of defects on tomatoes has been performed with transfer learning in [33]. The images for this detection were obtained after manual sorting based on different kinds of defects and are used for the transfer learning of the ResNet pre-trained on ImageNet dataset. The quality assessment of multiple kinds of apples including single-colour and multi-colour varieties has been performed with computer vision techniques in [34]. The randomness of grey-level pixels is used as a feature, where mean, variance, standard deviation, Root Mean Square (RMS), and Kurtosis were used for feature representation. The grey-level spatial variance was estimated by texture features. Both kinds of features are used as an input for a SVM and Sparse Representation Classifier (SRC) for the classification of defects in fruit. In another work, a combination of 18 colour and texture features has been used for grading tomatoes, where SVM has been used as a classifier [35].

2.3. Vision-Based Retail

Preliminary efforts related to the classification of fruit and vegetables at supermarket self-checkouts have been reported recently [36–40]. A MobileNet-based fruit classification system for a supermarket has been presented in [41]. A dataset of images of different fruit was obtained and used for transfer learning the MobileNet. The MobileNet architecture is selected to reduce the computational cost. To improve the overall effectiveness of MobileNet, new features are proposed as an input to MobileNet. A unique RGB code is defined for each fruit which is considered as a feature vector along with an RGB histogram and K-means centroid. An accuracy of 95% has been reported, however the number of varieties of fruit considered are significantly low. Considering the large number of fruit and vegetables sold at a supermarket, the proposed idea of a unique RGB code can be a limitation. The concept of using multiple patches of local features of a supermarket object was used in [42]. A Local Concepts Accumulation (LCA) layer is defined as a penultimate layer on CNN architecture. Entropy maximisation is used as a loss function for the classification of supermarket produce, where an accuracy of 100% has been achieved for ResNet with LCA. Recently, an attention fusion network has been proposed for image-based nutrition estimation of cooked food in [43]. A progressive weighted average of CNN weights is presented for the classification of fruit and vegetable images in [44]. Only colour and texture were considered as a feature for classification where a patch of 640 × 640 pixels was cropped from the images taken in a real supermarket environment. A more detailed discussion on utilisation of machine learning techniques for different applications including fruit and vegetables classification can be found in [41,45–47].

Current supermarket self-checkout systems require an unassisted selection from a LUT for billing of fruit and vegetables. This selection from the LUT can require good knowledge about the various species and kinds of fruit and vegetables, which increases the chances of an incorrect selection. The addition of a vision sensor can significantly improve the process of LUT-based selection. There can be many methods to realise for this application, for example a threshold can be set on the classification accuracy to consider it as a final selection. In the case where the classification accuracy is less than the threshold, the customer can be directed to a subset of the LUT with selections that are limited based on the classification results. The limited selection will be populated with a subset of similar fruit or vegetables varieties. This can significantly reduce the chances of incorrect selection and will improve the billing experience even if the systems cannot achieve 100% accuracy.

3. Data Acquisition and Pre-Processing

The working principles and apparatus design of supermarket self-checkouts have been studied in detail [36–40]. Considering the design of a supermarket self-checkout, we propose a prototype for acquiring images of fruit and vegetables, and for emulating the supermarket environment. The laboratory set-up for image and weight data acquisition and the organisation of obtained data are discussed below.

3.1. Prototype Design

The prototype consisted of multiple sensors for image acquisition, illuminance sensing, and weight sensing of individual fruit or vegetables. A detailed description of the multiple sensors used is presented in Table 2. A weight scale is used as a base for the placement of the fruit or vegetable and for the illuminance sensors. The relative positions of vision sensors are also considered from the centre of the weight scale. An AccuPost PP-70N was chosen as a low-cost weight sensor to obtain the weight of individual fruit and vegetable in the dataset. The scale has a resolution of 10 g and is easily compatible to multiple operating systems through a Universal Serial Bus (USB)-based connection. The supermarket environment involves significant challenges in terms of ambient lighting conditions. These ambient lighting conditions of a supermarket have been studied in detail, where an approximate illuminance level of 550–650 lux has been recommended in [48–52] for real-world supermarket environments

considering the required illuminance for the placement of items in shelves. A minimum illuminance of 500 lux has been recommended for trade counters i.e., self-checkout desks [48]. Considering this condition, we have used an illuminance of approximately (500–530) lux for image data acquisition. To measure the consistency of illuminance while taking images of fruit and vegetables a set of four Arduino BH1750 illuminance sensors (LS1, LS2, LS3, and LS4) was used. The incident ambient illuminance from a laboratory fluorescent ceiling light source on the weight scale and on fruits or vegetables placed at the centre of the scale was recorded. An Arduino Uno based on an ATmega-328 microcontroller was used for the integration of illuminance sensors and data acquisition with a USB-based connection. A detailed layout of the relative placement of the weight scale, light source, illuminance sensors, and the fruit or vegetable sample is described in Figure 1. Example illuminance values obtained with the sensors (LS1–LS4) are described in Table 3. These values are obtained by averaging the values for the first 500 samples per class. Two different vision sensors were used for image acquisition. The selection of sensors was made based on two considerations: (a) Using a readily-available low-cost embedded system [53] with High Definition (HD) cameras (e.g., ArduCAM, ArduinoCAM) and (b) using mobile phone cameras to support mobile platforms in future extensions of the proposed project. We have used ArduCAM (MT9F001) and Huawei P9 Lite mobile phone cameras as vision sensors for image acquisition. The vision sensors were mounted at a particular distance from the centre of the weight scale, considering the requirements of: (a) Capture of a reasonable area to accommodate fruit or vegetables that are significantly different in size, and (b) potential placement of vision sensor on a self-checkout in a supermarket. A detailed schematic of the vision sensors, illuminance sensors, and weight scale is provided for the experimental laboratory setup and a potential placement of a vision sensor on a self-checkout kiosk is illustrated in Figure 2.
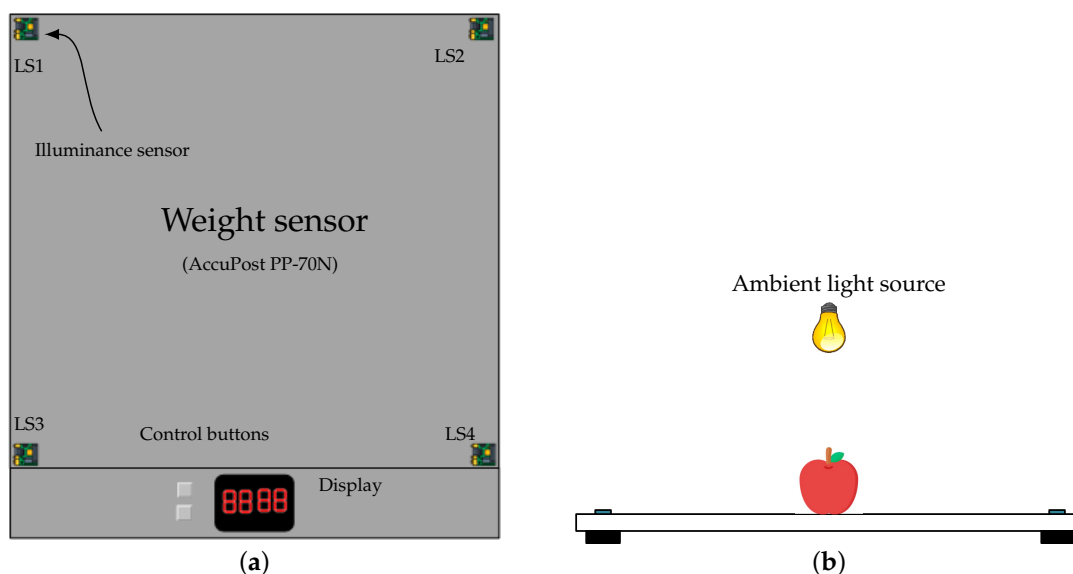

Figure 1. The detailed layout of: (**a**) Relative placement of weight scale, light source, illuminance sensor, and fruit or vegetable sample, and (**b**) Illustration of the light source w.r.t. fruit and illuminance sensors.

Table 2. Description of multiple sensors used for building the laboratory setup as a prototype of images based on supermarket self-checkouts for weight and image data acquisition.

	Vision Sensors				
	Brand Name	Resolution	Sensor	Height	Distance
1	ArduCAM MT9F001	4384 × 3288	1/2.3 inch CMOS	8 cm	19.5 cm
2	Huawei P9 Lite	3120 × 4160	Sony IMX214 Exmor RS	16 cm	30 cm
	Weight sensor				
3	AccuPost PP-70N	10 g–32 kg, USB 2.0/3.0 supported Windows 10			
	Illuminance sensor				
4	Ambient light sensor	Arduino BH1750 ambient light sensor			
	Controlling embedded system				
5	Embedded system	Arduino Uno (ATmega-328), 8-bit, 16 MHz			

Table 3. A description of the classes in the obtained dataset, nomenclature, and illuminance values obtained with sensors (LS1–LS4).

	Fruit/Vegetable	Nomenclature	Avg. Weight (kg)	Average Illuminance (LS1-LS4) Lux			
1	Brown onion	ONIBRXXXX	0.212	526.17	524.84	523.56	522.57
2	Carrot	CARROXXXX	0.064	525.02	527.20	522.95	524.39
3	Cauliflower	CABCAXXXX	0.419	526.34	525.12	525.35	523.08
4	Continental cucumber	CUCCOXXXX	0.014	533.08	525.70	529.26	525.94
5	Creme potato	POTCRXXXX	0.140	527.94	523.90	525.08	527.10
6	Drumhead cabbage	CABDRXXXX	0.833	534.03	528.22	523.58	529.09
7	Granny Smith apple	APPGSXXXX	0.164	523.79	522.65	525.75	526.06
8	Iceberg lettuce	LETICXXXX	0.432	531.46	525.87	530.42	526.30
9	Lady finger banana	BANLFXXXX	0.125	523.67	522.55	523.45	526.62
10	Mandarin	MANDAXXXX	0.138	525.85	526.88	529.81	526.49
11	Navel orange	ORANAXXXX	0.138	524.97	526.92	528.67	523.33
12	Packham pear	PEAPAXXXX	0.150	529.43	530.63	530.13	535.69
13	Pink lady apple	APPPLXXXX	0.326	529.32	523.56	522.13	535.77
14	Strawberry	BERSTXXXX	0.012	525.81	525.63	522.00	527.37
15	Tomato	TOMFIXXXX	0.132	523.55	523.56	526.91	532.98

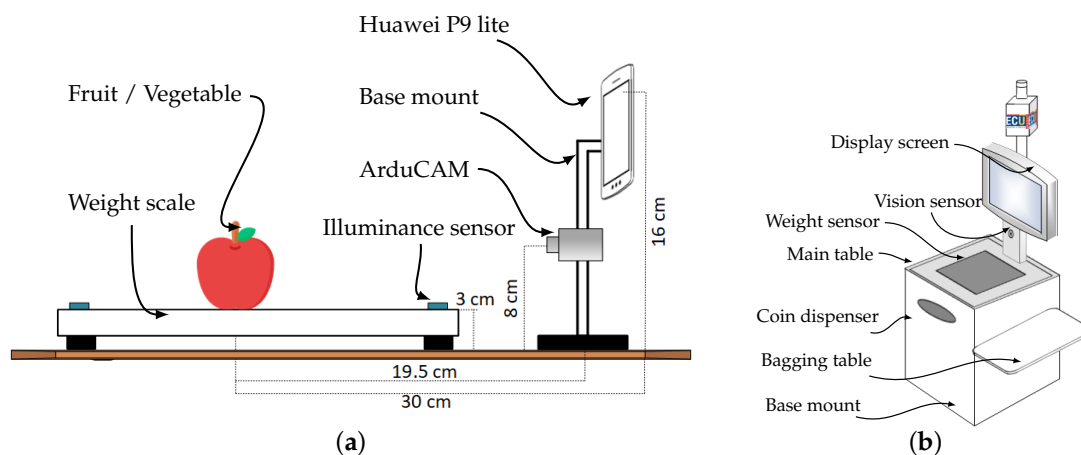XXXX: denotes the count of the samples per class.

Figure 2. A schematic of the proposed prototype: (**a**) Laboratory setup depicting the placement of vision sensor, illuminance sensors, weight sensor, and fruit or vegetable sample, and (**b**) A proposed placement of sensors on a typical supermarket self-checkout kiosk.

3.2. Image Acquisition

A dataset of fruit and vegetable images was obtained using the prototype laboratory setup based on considering the real-world supermarket environment. The prototype design was considered carefully to maintain the integrity among the images obtained with both vision sensors used. This integrity is important in order to use the obtained dataset for transfer learning of the CNN for classification, and maintaining classification effectiveness among the images of multiple sensors. The images of 15 different classes of fruit and vegetables were obtained where each class consists of 1000 images. The images were cropped to a maximum size of 3000 × 3000 pixels for both sensors, the initial resolution of obtained images is presented in Table 2 for both sensors. This image size was selected by considering the variations in sizes of fruit or vegetables used for building the dataset. The images were further ordered in a unified nomenclature along with the weight of individual fruit and vegetables saved in a separate repository. A description on the nomenclature and average weight of each class is presented in Table 3. Uniform ordering was achieved with the help of the nomenclature to integrate the weights and images in the dataset and to make the dataset consistent for future applications. Examples of obtained images are shown in Figure 3.
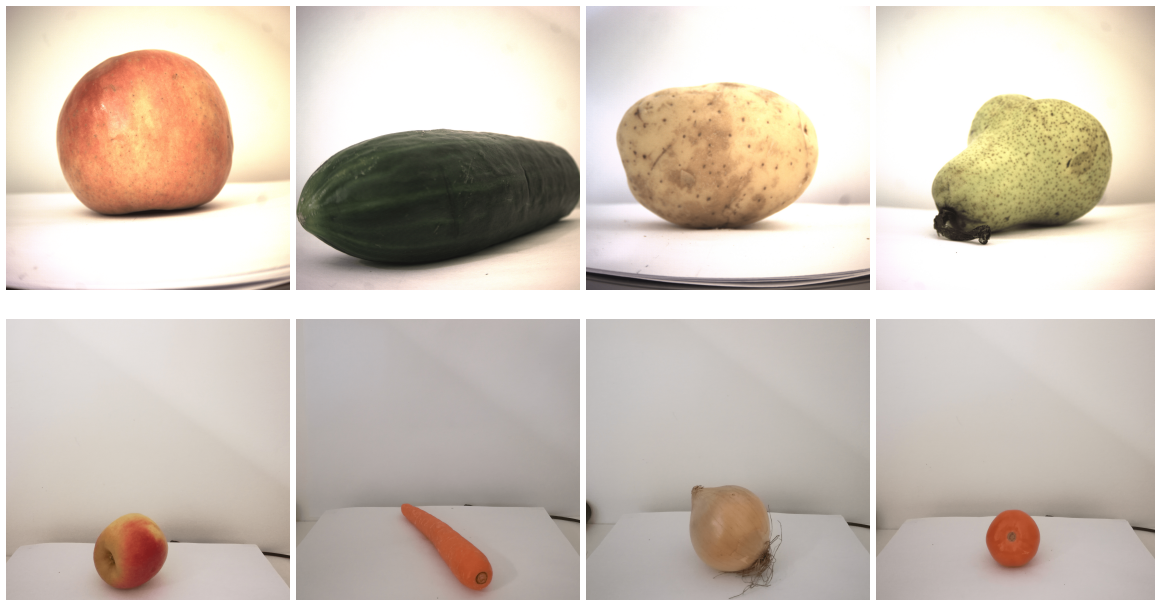

Figure 3. Example images obtained from the experiments with ArduCAM (**upper row**) and Huawei P9 Lite (**bottom row**).

4. Methodology

A coarse to fine classification-based two-stage classification technique is proposed in this paper. The fruit and vegetable images were initially classified into coarse classes, which are used to optimise a CNN for each coarse class to obtain the fine classification. A combined class level likelihood distribution is then estimated for the fine classification of all coarse classes so as to obtain the final classification described in Figure 4. This progressive classification is considered as a natural process where the weight is used as an inherent feature of the fruit and vegetable, which also helps in achieving better time complexity and memory requirements.

4.1. Coarse Classification

Initially, images are coarsely classified into three classes based on the weight information where the weight values are grouped into their natural distribution. The Jenks Natural Breaks (JNB) classification [54] technique is used to estimate the inherent natural distribution in the weight

information of the fruit and vegetable. The Accumulated Squared Deviation from the Mean (ASDM) of each class is reduced, and hence, the Accumulated Squared Deviation (ASD) among means of different classes is increased. A set of individual weights of each fruit or vegetable in a class i is represented as w_i, where the cardinality of w_i is considered as l_i. An integrated ordered vector of all weight sets of fruit and vegetable classes is denoted as:

$$W = \{w_1, w_2, w_3, \ldots, w_n\} = \{\varphi_1, \varphi_2, \varphi_3, \ldots, \varphi_m\}, \tag{1}$$

where n is the maximum number of classes and $m = \sum_{k=1}^{n} l_k$. To estimate the ASDM, a set of mean weights w.r.t. each class in W is represented as:

$$M = \{\mu_{w1}, \mu_{w2}, \mu_{w3}, \ldots, \mu_{wn}\}. \tag{2}$$

The accumulated deviation of individual weight value in w_i from mean μ_{wi} of a class i is estimated as:

$$\sigma_i^{ASDM} = \sum_{k=1}^{l_i} w_i(k) - \mu_{wi}, \tag{3}$$

where $\mu_{wi} = \frac{\sum_{k=1}^{l_i} w_i(k)}{l_i}$. The ASD among means of different classes is estimated based on W for all possible combination range distribution that can be described as:

$$\sigma_i^{ASD} = \sum_{j=1}^{n} \sum_{k=1}^{m} \varphi_k - \mu_i, \tag{4}$$

where the minimum value of σ_i^{ASD} represents the increased inter class deviation and hence the optimal distribution. A Goodness of Variance Fit (GVF) metric is maximised to estimate the effectiveness of the distribution. The GVF considered as a normalised difference of accumulated squared variance between class means and the weights of individual fruit and vegetables is described as:

$$GVF_i = \frac{(\sigma_i^{ASDM} - \sigma_i^{ASD})}{\sigma_i^{ASDM}}. \tag{5}$$

This is an iterative process where greater values of GVF indicates more effective distribution. This weight based coarse distribution groups the different varieties of a fruit or vegetable. This grouping helps in learning more effective features for the classification of the same species of a fruit or vegetable in the fine classification phase.
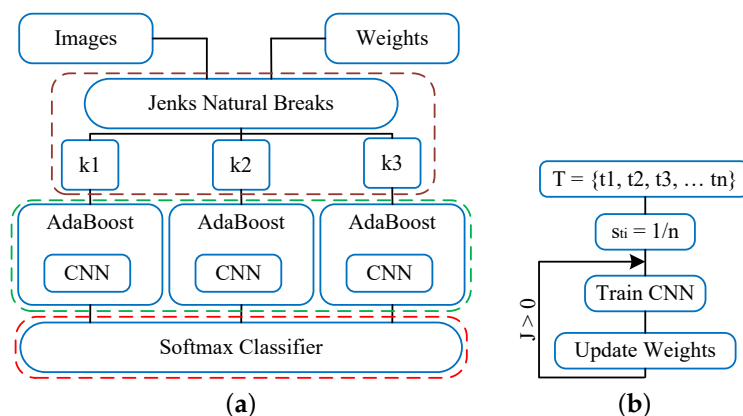

Figure 4. A stack based sequential description of the proposed approach: (**a**) A progressive coarse to fine classification based approach, and (**b**) AdaBoost CNN optimisation for fine classification of fruit and vegetables.

4.2. Fine Classification

A CNN has been optimised based on the AdaBoost [55] technique for each coarse class estimated by natural distribution. A sequential linear CNN boosting has been performed to obtain the classification results where a block level abstraction of coarse, fine, and final classification is presented in Figure 4.

Considering each coarse class as a combination of multiple classes, a multi-class classification problem can be defined as:

$$\widehat{c} = \arg \max_{\theta \in \{1...k\}} h_\theta(x), \tag{6}$$

where x is an unseen element of data randomly sampled from k classes. The classifier h_θ is trained on dataset $T = \{t_1, t_2, t_3, \ldots, t_n\}$ to assign a label $\widehat{c}$ to x such that the corresponding classification error is minimum. In our proposed approach, we have used the multi-class AdaBoost technique defined in [56] to optimise a CNN for each coarse class. The elements in the training dataset of each coarse class are initially weighted equally as: $s_{ti} = 1/n$ where n is the size of the training dataset. The CNN is then trained on T for J iterations to obtain an optimised CNN, where ImageNet weights have been used for the initialisation of the CNN when $J = 0$. The dataset weights of each element t_i are updated after each iteration for $J \geq 1$. The corresponding weight of each t_i is estimated by extracting a k dimensional classification likelihood vector with a trained classifier (i.e., CNN) $h_\theta^{j \geq 1}$. This k dimensional vector $\mathcal{P}$ is used for the estimation of weight for each t_i in T after every iteration, which can be described as:

$$s_{ti}^j = s_{ti}^{j-1} \, exp\left(- \alpha \, \frac{k-1}{k} \, \hat{c}_g^i \, \log(\mathcal{P}^j(t_i^{j-1})) \right), \tag{7}$$

where s_{ti} is the weight of ith training sample in T used in the jth iteration with a learning rate of α. The ground truth labels of corresponding classes are represented as $\widehat{c}_g^i$ for k dimensional likelihood vectors. The weights of the wrongly classified samples are improved in each iteration to optimise the classifier for wrongly classified samples in $j - 1$th iterations. The AdaBoost [56] uses a random forest as a combination of trees to make an ensemble of weak learners where, each contributing tree is initialised with random weights. The CNNs have the capability of finding a strong classification likelihood and correlations for a large dataset. However, considering the findings in (7) it can be concluded that strong correlation between $\widehat{c}_g^i$ and the output of the CNN will reduce the value of the exponential function, which will constrain the weights improvement to a small set of data not trained with the CNN previously. Training the CNNs on a small dataset can cause significant overfitting, and will add an overhead of extra computational cost. We initialised the CNN with the ImageNet weights for the first iteration where the weights of the CNN in further optimisation iterations have been retained and improved with the weighted training samples. This assumption has been made considering the sequential Representation Learning (RL) of a CNN in the training process, where retaining the previous information can help in the effective convergence of the CNN for a large dataset. This iterative process is repeated for all coarse classes obtained in the initial stage of the classification process. A detailed process for the fine classification stage has been described in Figure 4.

The CNNs used in the fine classification stage are a combination of a number of layers stacked together to perform the classification task. Each layer in the CNN plays a specific role for RL where the level of abstraction of the features learned increases from lower to upper layers. The low-level features i.e., pixel-level textures, are extracted with the help of conventional layers. These features are then combined in a Fully Connected (FC) layer. The flattened and combined representation obtained with the FC layer are then used to estimate the class level likelihood distribution for final classification. This class-level likelihood is estimated with the help of a softmax classifier. All these processes are performed sequentially for the classification task, a detailed description of the overall process of CNN-ased image classification can be found in [28]. The loss of feature representation learned in the training process of a CNN is propagated among the layers in each training step. A multi-class cross entropy-based loss is used in the proposed approach for the estimation of the discrete regression loss.

The AdaBoost optimisation based sample weights are considered at this stage for training on dataset T in a CNN, described as:

$$\mathcal{E}_i = \sum_{j=1}^{n} \hat{c}_g^j \log(t_i^j) \, s_{ti}^j, \tag{8}$$

where $\mathcal{E}_i$ is the cross entropy loss of training sample t_i. The corresponding ground truth label and sample weight are represented as $\hat{c}_g$ and s_{ti}, respectively.

4.3. Testing the Proposed Approach

The coarse to fine classification-based CNNs are trained for the classification of fruit and vegetables. To perform the classification of test images with the help of the proposed technique, the class likelihoods are obtained by each optimised CNN in the fine classification stage. The softmax layers of each CNN are removed to make a final classification, where a global softmax layer is added based on the concept in [57,58] represented as a bottom layer in Figure 4 and can be described as:

$$\sigma_{zi} = \frac{exp(z_i)}{\sum_{j=1}^{\Phi} exp(z_j)}, \tag{9}$$

where σ_{zi} is the normalised likelihood of an element z_i in a combined set of output probabilities obtained by fine classification CNNs. The combined number of classes are represented as Φ. These normalised probabilities obtained by the final softmax layer are used for the final classification of the fruit or vegetable sample to a class.

5. Implementation and Results

Experimental implementation and classification effectiveness achieved based on the dataset obtained in Section 3.2 is described in detail in this section. To validate our results a comparison of the proposed approach on GoogleNet, MobileNet, and a custom CNN, is performed.

5.1. Implementation

The experiments have been performed on the dataset obtained in Section 3. The images have been apportioned into 90%, 5%, and 5% segments for training, validation, and testing datasets, respectively. We used three CNNs with the proposed AdaBoost-based optimisation technique as base classifiers for implementation and testing. GoogleNet [58], MobileNet-v2 [59], and a 15-layer custom CNN based on the concept presented in [60] is used. A detailed description of the layers of the custom CNN is presented in Table 4. A decision was made to use a shallower network as compared to GoogleNet and MobileNet to optimise for the proposed technique. The Google and MobileNets were considered based on the assumption of a deeper and lighter weight CNN respectively to test our concept, where MobileNet is also intended to be used for mobile platforms with less computational power in our future extensions. Considering the small input image size of GoogleNet and MobileNet, we considered a larger input image size in the custom CNN. The custom CNN consists of a sequential combination of convolutional (Conv), pooling and Fully Connected (FC) layers followed by a softmax classification layer to estimate the class-level probability distribution. Considering the capabilities of sparse representation and equivariant parameters sharing we used a sequence of convolutional and pooling layers for RL. The architecture used for the custom CNN has been reported as state-of-the-art in comparison to logistic regression, Extreme Learning (EL), and SVM in [60]. The local features of a fruit or vegetable image are extracted by the application of a convolution operation with particular kernel size and number of nodes as described in Table 4. A ReLU function is applied as a threshold on the features obtained from the convolutional nodes where filtered features are represented as the output of the layer. The neighbouring statistical summary of the features is extracted and converted to an invariant representation with the help of a pooling operation applied to the output of the convolutional layers. The depth of the custom CNN is considered carefully in comparison to the

Google and MobileNets where the custom CNN is considered as a weak classifier for optimisation with the proposed AdaBoost technique. The ReLU was used as an activation function for all hidden layers where the weighted training sample based cross entropy loss defined in (8) was used for training. Experiments have been performed on a 12 GB Tesla K80 with 32 GB of installed memory.

Table 4. Description of the custom CNN used as the base classifier in the AdaBoost optimisation based on [60].

	Layer	Kernel Size	No. of Nodes	Stride	Padding	Layer Weights	Layer Bias	Output Size
1	Input	-	-	-	-	-	-	$512 \times 512 \times 3$
2	Conv	7×7	40	3×3	0×0	$7 \times 7 \times 3 \times 40$	$1 \times 1 \times 40$	$170 \times 170 \times 40$
3	Pooling	3×3	-	3×3	0×0	-	-	$56 \times 56 \times 40$
4	Conv	7×7	80	3×3	2×2	$7 \times 7 \times 40 \times 80$	$1 \times 1 \times 80$	$18 \times 18 \times 80$
5	Pooling	3×3	-	1×1	1×1	-	-	$18 \times 18 \times 80$
6	Conv	3×3	120	1×1	1×1	$3 \times 3 \times 80 \times 120$	$1 \times 1 \times 120$	$18 \times 18 \times 120$
7	Pooling	3×3	-	1×1	1×1	-	-	$18 \times 18 \times 120$
8	Conv	3×3	80	1×1	1×1	$3 \times 3 \times 120 \times 80$	$1 \times 1 \times 80$	$18 \times 18 \times 80$
9	Pooling	3×3	-	1×1	1×1	-	-	$18 \times 18 \times 80$
10	Conv	1×1	80	1×1	1×1	$1 \times 1 \times 80 \times 80$	$1 \times 1 \times 80$	$20 \times 20 \times 80$
11	Pooling	3×3	-	1×1	1×1	-	-	$20 \times 20 \times 80$
12	FC	-	40	-	-	-	-	$1 \times 1 \times 40$
13	FC	-	15	-	-	-	-	$1 \times 1 \times 15$
14	Softmax	-	-	-	-	-	-	$1 \times 1 \times 15$
15	Output	-	-	-	-	-	-	$1 \times 1 \times 15$

Conv: represents the convolutional layer, FC: represents the fully connected layer.

5.2. Experimental Results

The experiments were performed with all three CNNs i.e., GoogleNet, MobileNet, and the custom CNN. The classification results obtained with the transferred learned pre-trained GoogleNet and MobileNet were used for comparison. The Google and MobileNets were initialised with the ImageNet weights where we used Xavier's initialisation [61] technique was used for the initialisation of the custom CNN. A weight-based coarse classification was performed based on the JNB technique defined in ((4) and (5)). The result based on the weight-based classification is shown in Table 5 for a GVF of 0.65. This GVF was selected based on the experimental results obtained, where an approximately equal size of classes was considered for coarse classification. However, the AdaBoost technique is considered significantly prone to imbalanced class sizes [55].

Table 5. Jenks Natural Break (JNB) based coarse classification of fruit and vegetables.

	Fruit/Vegetables	Avg. Weight (kg)	Weight. Dev	Class	% of Dataset
1	Strawberry	0.012	0.02		
2	Continental cucumber	0.014	0.07		
3	Carrot	0.064	0.02		
4	Lady finger banana	0.125	0.02	Class 1	46.66
5	Tomato	0.132	0.02		
6	Mandarin	0.138	0.03		
7	Navel orange	0.138	0.03		
8	Creme potato	0.140	0.03		
9	Packham pear	0.150	0.03	Class 2	26.66
10	Granny Smith apple	0.164	0.03		
11	Brown onion	0.212	0.04		
12	Pink lady apple	0.326	0.19		
13	Cauliflower	0.419	0.05	Class 3	26.66
14	Iceberg lettuce	0.432	0.03		
15	Drumhead cabbage	0.833	0.06		

GoogleNet was considered as a deep base classifier for AdaBoost optimisation. The accuracy attained for the training and test datasets is presented in Table 6 for different epochs, where samples were randomly selected and shuffled for both datasets. The training and test accuracy are proportional for the initial 12 epochs, however the accuracy of the test set decreases for higher numbers of epochs. The basic intuition of AdaBoost was to use a linear combination of weak classifiers [55], GoogleNet in comparison is a deep classifier that can approximate the strong correlations. Hence, using GoogleNet with AdaBoost for a higher number of epochs increases bias for the test set. MobileNet was considered as a light weight CNN for AdaBoost-based optimisation to classify fruit and vegetables. The accuracy of MobileNet is presented in Table 6 for multiple epochs. The test accuracy of MobileNet increases for the first 12 epochs and remains consistent up to 15 epochs however the accuracy deceases when 18–20 epochs are used. For a higher number of epochs, the AdaBoost technique assigns negligible weights to the correctly classified samples, so to improve the weights of wrongly-classified samples. This negligible weight assignment causes a significant bias for the partial training dataset. This bias causes an overfitting for higher numbers of epochs and hence, a decrease in the classification accuracy of the test dataset. This decrease is due to partial training of the CNNs after a particular number of epochs, which depends upon the size and number of parameters in the CNN. This partial training can be considered a kind of overfitting where training a CNN on (a) a small dataset, and (b) higher number of epochs increases the CNN bias for unseen test samples. The classification accuracy for AdaBoost optimisation of the Google and MobileNets is compared with the transfer learned pre-trained Google and Mobile Nets on the ImageNet dataset. To transfer learn, a set of 500 images per class was used for training, where both CNNs were trained for 30 epochs. A set of 250 images per class was used for cross validation in the transfer learning phase.

The custom CNN is considered a weak learner for AdaBoost optimisation in the proposed technique. The CNN consists of 15 layers that are based on the architecture proposed in [60]. The custom CNN was trained for 25 epochs, where the result for multiple epochs is described in Table 7. A similar CNN test accuracy trend has been noted however, the custom CNN is less prone to negligible weight criteria observed for both the Google and MobileNets. A significant conclusion can be drawn here that the AdaBoost-based optimisation of CNNs can converge to complex data correlations with smaller or less deep networks. This makes the proposed approach more suitable for larger and complex correlations in datasets, i.e., classification of different varieties of a fruit or vegetable with less computational requirements. Moreover, weight-based coarse classification used in the proposed approach also helps in reducing the computational and memory requirements. A detailed comparison of the confusion matrix-based classification metrics of accuracy, Error Rate (ER), Positive Predictive Value (PPV), True Negative Rate (TNR), True Positive Rate (TPR), and F1 score is presented in Table 8. The classification accuracy of each class is obtained as a ratio of correctly classified images and the total number of images of a class, where the effectiveness of the proposed techniques is presented as ER. The precision or PPV is presented as the ratio of correctly predicted images and the total number of images identified as a particular fruit or vegetable. Test accuracy is also presented as an F1 score, which is obtained as a harmonic mean of precision and recall. The proposed approach can be considered significantly prone to complex and imbalanced dataset distributions. This implication can be observed by the average TPR or sensitivity and the F1 score (93.57%) that is comparable to the overall accuracy presented in Table 7. It can be observed that approximately 11 out of 15 fruit or vegetables can be classified with an accuracy of 99%. A classification confusion matrix of the custom CNN AdaBoost optimisation is depicted in Figure 5 for the fruit and vegetable classes presented in Table 3.

An inference time analysis was performed to estimate the practical implementation of the proposed technique. A batch of 15 random images, with one from each class was selected for inference analysis where the total inference time (ms) was noted as the time to classify all 15 images. We performed this analysis on a device for both CPU and GPU-based classification where the fastest CPU-based inference is approximately three times slower than GPU-based inference. A description of

the hardware used for the computation is presented in Table 9. The images were loaded in the form of a tensor in the memory where total inference time includes a time to read the tensor of 15 images from memory and the model computation time. On average, a GPU-based inference of an image takes approximately 588.44 ms which is 2.8 times faster than the CPU-based inference time of 1647.65 ms with the optimised custom CNN. A comparison of inference times for the AdaBoost optimised Google, MobileNets, and custom CNN models is presented in Table 10, the time for single image inference is obtained by dividing the total inference time by the number of images in the batch. The inference time for Google and MobileNets is significantly higher than the proposed AdaBoost-optimised CNN network, however, an inference in a real implementation will also depends upon the Input/Output (I/O) and related overheads of an execution platform.

Table 6. A comparison of training and test accuracy of transfer learned and AdaBoost-optimised Google and MobileNets.

Epochs	Network	Training Accuracy (%)	Test Accuracy (%)	Network	Training Accuracy (%)	Test Accuracy (%)
10		81.90	78.56		78.69	71.52
15		93.45	82.71		81.23	78.86
20	Pre-trained GoogleNet	94.65	81.78	Pre-trained MobileNet	89.98	80.23
25		96.45	**83.56**		94.87	81.44
30		**95.67**	82.10		**95.56**	**83.15**
10		81.86	72.96		86.56	81.45
12		87.10	**81.24**		92.63	87.21
14	AdaBoost GoogleNet	89.74	78.58	AdaBoost MobileNet	94.44	88.45
16		93.25	76.63		**95.50**	**91.33**
18		**96.21**	76.00		94.88	87.56

Table 7. Training and test accuracy achieved with AdaBoost-optimised custom CNN.

Epochs	Training Accuracy (%)	Test Accuracy (%)
10	93.10	80.13
15	94.17	83.43
20	96.42	88.69
22	95.67	**93.97**
25	**97.14**	85.11

Table 8. Classification metric comparison for AdaBoost-optimised CNN based fruit and vegetables classification.

	Fruit/Vegetable	Accuracy (%)	ER (%)	PPV (%)	TNR (%)	TPR (%)	F1 Score
1	Brown onion	99.47	0.53	96.00	99.71	96.00	0.960
2	Carrot	99.73	0.27	98.00	99.86	98.00	0.980
3	Cauliflower	99.47	0.53	100.00	100.00	92.00	0.958
4	Continental cucumber	99.73	0.27	100.00	100.00	96.00	0.980
5	Creme potato	99.07	0.93	90.57	99.29	96.00	0.932
6	Drumhead cabbage	98.40	1.60	85.19	98.86	92.00	0.885
7	Granny Smith apple	99.47	0.53	96.00	99.71	96.00	0.960
8	Iceberg lettuce	98.53	1.47	88.24	99.14	90.00	0.891
9	Lady finger banana	99.87	0.13	100.00	100.00	98.00	0.990
10	Mandarin	97.73	2.27	83.67	98.86	82.00	0.828
11	Navel orange	97.60	2.40	80.77	98.57	84.00	0.824
12	Packham pear	99.60	0.40	97.96	99.86	96.00	0.970
13	Pink lady apple	99.60	0.40	100.00	100.00	94.00	0.969
14	Strawberry	99.60	0.40	97.96	99.86	96.00	0.970
15	Tomato	99.70	0.93	90.57	99.29	96.00	0.932

	Brown onion	Carrot	Cauliflower	Continental cucumber	Creme potato	Drumhead cabbage	Granny Smith apple	Iceberg lettuce	Lady finger banana	Mandarin	Navel orange	Packham pear	Pink lady apple	Strawberry	Tomato
Brown onion	97.31	0.00	0.00	0.00	2.69	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Carrot	0.00	98.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.00
Cauliflower	0.00	0.00	93.65	0.00	2.60	0.00	0.00	3.75	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Continental cucumber	0.00	0.00	0.00	97.00	0.00	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Creme potato	3.20	0.00	0.00	0.00	96.80	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Drumhead cabbage	0.00	0.00	0.00	0.00	0.00	93.00	0.00	7.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Granny Smith apple	0.00	0.00	0.00	0.00	0.00	2.00	96.00	0.00	0.00	0.00	0.00	2.00	0.00	0.00	0.00
Iceberg lettuce	0.00	0.00	0.00	0.00	0.00	11.00	0.00	89.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Lady finger banana	0.00	0.00	0.00	0.00	2.00	0.00	0.00	0.00	98.00	0.00	0.00	0.00	0.00	0.00	0.00
Mandarin	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	82.00	18.00	0.00	0.00	0.00	0.00
Navel orange	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	16.00	84.00	0.00	0.00	0.00	0.00
Packham pear	0.00	0.00	0.00	0.00	0.00	0.00	4.00	0.00	0.00	0.00	0.00	96.00	0.00	0.00	0.00
Pink lady apple	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	95.00	0.00	5.00
Strawberry	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	98.00	1.00
Tomato	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	3.00	96.00

Figure 5. A classification confusion matrix of fruit and vegetables for the custom CNN.

Table 9. Hardware description for inference time analysis.

	Device Type	Memory	Execution Unit
1	CPU	16 GB	Intel Xenon (8-cores)
2	GPU	32 GB	Tesla K80 (4992-cores)

Table 10. Inference time (per image) comparison for proposed approach.

	Model	CPU (ms)	GPU (ms)
1	GoogleNet	1954.32	723.82
2	MobileNet	1889.56	674.84
3	Custom CNN	1647.65	588.44

6. Conclusions

The classification of fruit and vegetables includes significant challenges due to the highly variable physical features of a fruit or vegetable which can include shape, size, colour, texture, and level of ripeness. On top of this, the classification of fruit and vegetables at supermarket checkouts faces additional challenges due to ambient lighting conditions and human factors. In this paper, we proposed a progressive coarse to fine classification-based technique for classifying fruit and vegetables at supermarket self-checkouts. The weight of individual fruit or vegetable was used

for coarse classification from 15 classes down to three using the Jenks Natural Breaks classification technique. These three classes are then used for AdaBoost-based optimisation of CNNs for fine classification. The training samples were initially weighted equally and their weights then improved in each iteration to optimise the CNN, where samples from the wrongly classified classes were weighted more as compared to other classes. The results obtained from all three fine classification CNNs were then used to estimate a multi-class probability distribution for final classification. Three kinds of CNNs were used for comparing and testing the proposed technique. GoogleNet, MobileNet-v2, and a custom 15-layer CNN were used based on the following criteria: (a) Selection of a deep CNN for optimisation with the proposed technique, (b) selection of a light weight small CNN for optimisation, and (c) selection of a weak classifier for optimisation. The experiments were performed for all three CNNs and a positive result was obtained for all three CNNs, where the custom CNN-based weak classifier was considered the most effective despite a lower number of parameters and computational requirements. Considering the capability of the proposed approach to classify the complex data correlations i.e., classification of different kinds of fruit and vegetables, this approach looks promising for applications to large datasets in a real supermarket environment.

References

1. Nakos, N. Coles Market Share in Australia Has Declined. Australian Food News. Available online: https://www.ausfoodnews.com.au/2017/10/18/coles-market-share-in-australia-has-declined.html (accessed on 13 November 2020).
2. Hogan, A. Supermarkets Dominate IBISWorld Top 1000 Australian Companies List. Available online: https://www.ausfoodnews.com.au/2017/03/10/supermarkets-dominate-ibisworld-top-1-000-australian-companies-list.html (accessed on 13 November 2020).
3. O'Farrell, K. *Australian Plastics Recycling Survey National Report*; Department of Environment and Energy Australia: Sydney, Australia, 2017.
4. Geyer, R.; Jambeck, J.R.; Law, K.L. Production, use, and fate of all plastics ever made. *Sci. Adv.* **2017**, *3*, e1700782. [CrossRef] [PubMed]
5. Herwig, N.C. Method and Apparatus for Reducing Recognition Times in an Image-Based Product Recognition System. U.S. Patent 9,135,789, 15 September 2015.
6. Walter, J.; Morrison, J.; Lin, H.J. Self-Checkout System. U.S. Patent 6,990,463, 24 January 2006.
7. Iizuka, H. Information Processing Apparatus and Print Control Method. U.S. Patent 8,553,251, 8 October 2013.
8. Dhankhar, M. Automated Object Recognition Kiosk for Retail Checkouts. U.S. Patent 10,366,445, 30 July 2019.
9. Chung, C.L.; Huang, K.J.; Chen, S.Y.; Lai, M.H.; Chen, Y.C.; Kuo, Y.F. Detecting Bakanae disease in rice seedlings by machine vision. *Comput. Electron. Agric.* **2016**, *121*, 404–411. [CrossRef]
10. Ganganagowder, N.V.; Kamath, P.R. Intelligent classification models for food products basis on morphological, colour and texture features. *Acta Agronómica* **2017**, *66*, 486–494. [CrossRef]
11. Sun, Y.; Gu, X.; Sun, K.; Hu, H.; Xu, M.; Wang, Z.; Tu, K.; Pan, L. Hyperspectral reflectance imaging combined with chemometrics and successive projections algorithm for chilling injury classification in peaches. *Lwt* **2017**, *75*, 557–564. [CrossRef]
12. Zhang, J.; Wang, N.; Yuan, L.; Chen, F.; Wu, K. Discrimination of winter wheat disease and insect stresses using continuous wavelet features extracted from foliar spectral measurements. *Biosyst. Eng.* **2017**, *162*, 20–29. [CrossRef]

13. Liu, S.; Cossell, S.; Tang, J.; Dunn, G.; Whitty, M. A computer vision system for early stage grape yield estimation based on shoot detection. *Comput. Electron. Agric.* **2017**, *137*, 88–101. [CrossRef]

14. Fernández, R.; Montes, H.; Surdilovic, J.; Surdilovic, D.; Gonzalez-De-Santos, P.; Armada, M. Automatic Detection of Field-Grown Cucumbers for Robotic Harvesting. *IEEE Access* **2018**, *6*, 35512–35527. [CrossRef]

15. Altaheri, H.; Alsulaiman, M.; Muhammad, G. Date Fruit Classification for Robotic Harvesting in a Natural Environment Using Deep Learning. *IEEE Access* **2019**, *7*, 117115–117133. [CrossRef]

16. SepúLveda, D.; Fernández, R.; Navas, E.; Armada, M.; González-De-Santos, P. Robotic Aubergine Harvesting Using Dual-Arm Manipulation. *IEEE Access* **2020**, *8*, 121889–121904. [CrossRef]

17. Birrell, S.; Hughes, J.; Cai, J.Y.; Iida, F. A field-tested robotic harvesting system for iceberg lettuce. *J. Field Robot.* **2020**, *37*, 225–245. [CrossRef]

18. Yamamoto, K.; Guo, W.; Yoshioka, Y.; Ninomiya, S. On plant detection of intact tomato fruits using image analysis and machine learning methods. *Sensors* **2014**, *14*, 12191–12206. [CrossRef] [PubMed]

19. Li, B.; Long, Y.; Song, H. Detection of green apples in natural scenes based on saliency theory and Gaussian curve fitting. *Int. J. Agric. Biol. Eng.* **2018**, *11*, 192–198. [CrossRef]

20. Barnea, E.; Mairon, R.; Ben-Shahar, O. Colour-agnostic shape-based 3D fruit detection for crop harvesting robots. *Biosyst. Eng.* **2016**, *146*, 57–70. [CrossRef]

21. Bhargava, A.; Bansal, A. Fruits and vegetables quality evaluation using computer vision: A review. *J. King Saud-Univ. Comput. Inf. Sci.* **2018**, *1*, 1–15. [CrossRef]

22. Zhang, H.; Wu, J.; Zhao, Z.; Wang, Z. Nondestructive firmness measurement of differently shaped pears with a dual-frequency index based on acoustic vibration. *Postharvest Biol. Technol.* **2018**, *138*, 11–18. [CrossRef]

23. Rachmawati, E.; Supriana, I.; Khodra, M.L. Toward a new approach in fruit recognition using hybrid RGBD features and fruit hierarchy property. In Proceedings of the International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), Yogyakarta, Indonesia, 19–21 September 2017; pp. 1–6. [CrossRef]

24. Zhang, Z.; Cao, R.; Peng, C.; Liu, R.; Sun, Y.; Zhang, M.; Li, H. Cut-Edge Detection Method for Rice Harvesting Based on Machine Vision. *Agronomy* **2020**, *10*, 590. [CrossRef]

25. Chen, M.; Tang, Y.; Zou, X.; Huang, K.; Huang, Z.; Zhou, H.; Wang, C.; Lian, G. Three-dimensional perception of orchard banana central stock enhanced by adaptive multi-vision technology. *Comput. Electron. Agric.* **2020**, *174*, 105508. [CrossRef]

26. Tang, Y.C.; Wang, C.; Luo, L.; Zou, X.; Chen, M.; LI, J. Recognition and localization methods for vision-based fruit picking robots: A review. *Front. Plant Sci.* **2020**, *11*, 510. [CrossRef]

27. Tian, H.; Wang, T.; Liu, Y.; Qiao, X.; Li, Y. Computer vision technology in agricultural automation A review. *Inf. Process. Agric.* **2020**, *7*, 1–19. [CrossRef]

28. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*; The MIT Press: Cambridge, MA, USA, 2012; pp. 1097–1105.

29. Nasiri, A.; Taheri-Garavand, A.; Zhang, Y.D. Image-based deep learning automated sorting of date fruit. *Postharvest Biol. Technol.* **2019**, *153*, 133–141. [CrossRef]

30. Farooq, M.; Sazonov, E. Feature extraction using deep learning for food type recognition. In Proceedings of the International Conference on Bioinformatics and Biomedical Engineering (ICBBE), Seoul, Korea, 12–14 November 2017; pp. 464–472._41. [CrossRef]

31. Srivastava, S.; Vani, B.; Sadistap, S. Machine-vision based handheld embedded system to extract quality parameters of citrus cultivars. *J. Food Meas. Charact.* **2020**, *14*, 2746–2759. [CrossRef]

32. Habib, M.T.; Majumder, A.; Jakaria, A.; Akter, M.; Uddin, M.S.; Ahmed, F. Machine vision based papaya disease recognition. *J. King Saud-Univ. Inf. Sci.* **2020**, *32*, 300–309. [CrossRef]

33. Zanatta da Costa, A.; da Costa, A.Z.; Figueroa, H.E.; Fracarolli, J.A. Computer vision based detection of external defects on tomatoes using deep learning. *Biosyst. Eng.* **2020**, *190*, 131–144. [CrossRef]

34. Bhargava, A.; Bansal, A. Quality evaluation of Mono & Bi-Colored Apples with computer vision and multispectral imaging. *Multimed. Tools Appl.* **2020**, *79*, 7857–7874. [CrossRef]

35. Kumar, S.D.; Esakkirajan, S.; Bama, S.; Keerthiveena, B. A Microcontroller based Machine Vision Approach for Tomato Grading and Sorting using SVM Classifier. *Microprocess. Microsyst.* **2020**, *76*, 103090. [CrossRef]

36. Femling, F.; Olsson, A.; Alonso-Fernandez, F. Fruit and Vegetable Identification Using Machine Learning for Retail Applications. In Proceedings of the International Conference on Signal-Image Technology

Internet-Based Systems (SITIS), Las Palmas de Gran Canaria, Spain, 26–29 November 2018; pp. 9–15. [CrossRef]

37. Hossain, M.S.; Al-Hammadi, M.; Muhammad, G. Automatic Fruit Classification Using Deep Learning for Industrial Applications. *IEEE Trans. Ind. Inform.* **2019**, *15*, 1027–1034. [CrossRef]

38. Licht, Y.Z.; Saker, R.D. Reinforcement Machine Learning for Item Detection. U.S. Patent 20,200,042,491, 6 February 2020.

39. Schögel, M.; Lienhard, S.D. Cashierless Stores the New Way to the Customer. *Mark. Rev. St. Gall.* **2020**, *30*, 1–5.

40. Patil, A.R.; Paolella, M.; Palella, M.; Trivelpiece, S.E. Self-Service Product Return Using Computer Vision and Artificial Intelligence. U.S. Patent 20,200,151,735, 14 May 2020.

41. Rojas-Aranda, J.L.; Nunez-Varela, J.I.; Cuevas-Tello, J.; Rangel-Ramirez, G. Fruit Classification for Retail Stores Using Deep Learning. In *Pattern Recognition, Mexican Conference on Pattern Recognition (MCPR)*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 3–13.

42. Srivastava, M.M. Bag of Tricks for Retail Product Image Classification. In *Image Analysis and Recognition*; Springer: Berlin, Germany, 2020; pp. 71–82.

43. Liu, C.; Liang, Y.; Xue, Y.; Qian, X.; Fu, J. Food and Ingredient Joint Learning for Fine-Grained Recognition. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *1*, 1051–8215. [CrossRef]

44. Hameed, K.; Chai, D.; Rassau, A. A progressive weighted average weight optimisation ensemble technique for fruit and vegetable classification. In Proceedings of the International Conference on Control, Automation, Robotics and Vision (ICARCV), Shenzhen, China, 13–15 December 2020; pp. 1–6.

45. Hameed, K.; Chai, D.; Rassau, A. A comprehensive review of fruit and vegetable classification techniques. *Image Vis. Comput.* **2018**, *80*, 24 – 44. [CrossRef]

46. Rehman, T.U.; Mahmud, M.S.; Chang, Y.K.; Jin, J.; Shin, J. Current and future applications of statistical machine learning algorithms for agricultural machine vision systems. *Comput. Electron. Agric.* **2019**, *156*, 585–605. [CrossRef]

47. Tripathi, M.K.; Maktedar, D.D. A role of computer vision in fruits and vegetables among various horticulture products of agriculture fields: A survey. *Inf. Process. Agric.* **2020**, *7*, 183–203. [CrossRef]

48. Recommended Lighting Levels. Available online: https://decrolux.com/news/2017/recommended-lighting-levels (accessed on 18 September 2020).

49. Grocery Store Lighting Guide for Making Food Look Fresh. Available online: https://www.standardpro.com/grocery-store-lighting/ (accessed on 18 September 2020).

50. Light Level Recommendations for Safe, Healthy & Comfortable Lighting. Available online: https://www.rexellighting.co.nz/uploads/attachments/Light-Level-Recommendations.pdf (accessed on 18 September 2020).

51. Supermarket Lighting Design Guide. Available online: https://www.contechlighting.com/en/docs/contechsupermarketlightingguide2018_0.pdf (accessed on 18 September 2020).

52. Quartier, K.; Christiaans, H.; Van Cleempoel, K. Retail design: Lighting as an atmospheric tool, creating experiences which influence consumers' mood and behaviour in commercial spaces. In Proceedings of the Design Research Society Conference (DRSC), Sheffield, UK, 16–19 July 2008; pp. 1–17.

53. Alvi, M.B.; Hameed, K.; Alvi, M.; Javed, W.; Afzal, M. Algorithmic State Machine and Data Based Modeling of Superscalar Processor of Order 2. In Proceedings of the International Conference on Software Technology and Engineering (ICSTE), Kuala Lumpur, Malaysia, 12–14 August 2011; pp. 1–5. [CrossRef]

54. Jenks, G. *Optimal Data Classification for Choropleth Maps Occasional Paper No. 2*; Department of Geography, University of Kansas: Lawrence, Kansas, 1977.

55. Freund, Y.; Schapire, R.; Abe, N. A short introduction to boosting. *J.-Jpn. Soc. Artif. Intell.* **1999**, *14*, 1612. [CrossRef]

56. Hastie, T.; Rosset, S.; Zhu, J.; Zou, H. Multi-class adaboost. *Stat. Interface* **2009**, *2*, 349–360. [CrossRef]

57. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014; pp. 818–833._53. [CrossRef]

58. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9. [CrossRef]

59. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520. [CrossRef]

60. Zhang, Y.D.; Dong, Z.; Chen, X.; Jia, W.; Du, S.; Muhammad, K.; Wang, S.H. Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation. *Multimed. Tools Appl.* **2017**, *78*, 3613–3632. [CrossRef]

61. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the International Conference on Artificial Intelligence and Statistics (ICAIS), Las Vegas, NV, USA, 12–15 July 2010; pp. 249–256. [CrossRef]