Knowledge Engineering and Data Science (KEDS)
Vol 7, No 1, April 2024, pp. 117–127

pISSN 2597-4602
eISSN 2597-4637

A Novel Approach to Defect Detection in Arabica Coffee Beans Using Deep Learning: Investigating Data Augmentation and Model Optimization

Yusriel Ardian [a,1], Novta Danyel Irawan [a,2], Sutoko [a,3], I Nyoman Gede Arya Astawa [b,4,*], Ida Bagus Irawan Purnama [c,5], Felix Andika Dwiyanto [d,6]

[a] *Department of Information Technology, Politeknik Unisma Malang*
Jalan Mayjen Haryono No.193, Malang 65144, Indonesia
[b] *Department of Information Technology, Politeknik Negeri Bali*
Kampus Jimbaran, Badung, Bali, 80361 Indonesia
[c] *Department of Electrical Engineering, Politeknik Negeri Bali*
Kampus Jimbaran, Badung, Bali, 80361 Indonesia
[d] *Faculty of Computer Science, AGH University of Krakow*
al. Adama Mickiewicza 30, Krak'ow 30-059, Poland
[1] *yusriel@polisma.ac.id;* [2] *novta@polisma.ac.id;* [3] *sutoko@polisma.ac.id;* [4] *astawa@pnb.ac.id*;* [5] *ida.purnama@pnb.ac.id;*
[6] *dwiyanto@agh.edu.pl*
**corresponding author*

ARTICLE INFO

Article history:
Received 26 August 2024
Revised 29 September 2024
Accepted 23 October 2024
Published online 31 October 2024

Keywords:
Arabica coffee beans
Defect detection
CNN
VGG16 architecture
Data augmentation and optimization

ABSTRACT

Arabica coffee beans have valuable market worth because of their taste and quality, and there are defects like wholly and partially black beans that can lower the standards of a product, especially in the premium coffee sector. However, the manual processes used to detect the defects take an inordinate amount of time and are inefficient. This study aims to bridge the knowledge gap on the automated detection and recognition of the defects present in the Arabica coffee beans by creating and optimizing a CNN model based on a modified VGG16 architecture. The model applies data augmentation, rotation, cropping, and Bayesian hyperparameter optimization to improve defect detectability and expedite the training period. During testing, the defined model demonstrated excellent efficiency in defect detection, with a 97.29% confidence level, which was higher than that of the modified VGG16 and Slim-CNN models. The goal of the second optimization was an improvement of the practical application of the model. In terms of the time it takes for a model to be trained, approximately 30% of the time was saved. These findings present a consistent and effective way for the mass production processes of coffee to have quality control procedures automated. The model's ability to detect defects in other agricultural items makes it attractive, thus serving as a practical example of how AI can impact effective management in the inspection processes. The research further enriches the study of deep learning applications in agriculture by demonstrating how to efficiently address specific defect detection problems through an optimized convolutional neural network model.

I. Introduction

Arabica coffee bean quality determines the flavor [1], aroma [2], and money the product can bring to its producers [3], especially those targeting the high-end market. Entirely black and partially black beans as defects affect not only the sensory attributes of coffee but also the marketing around it, its pricing, and exploitation chances [4]. In these markets, producers need to maintain consumer loyalty and keep making profits by ensuring high-quality coffee beans [5]. As a result, the appropriate identification of defects in embryos and other rudiments is an essential stage for the productivity and standardization of coffee enough for commercial purposes.

In the past, the emphasis of defect detection processes was placed on human beings, which is manual [6]. This forms limitations, such as human judgment errors, too much time required, and even while effort is wasted sorting many beans [7]. However, will some automated monitoring solutions be introduced to speed up the process? With the current trends towards mass production and increased demand, manual inspection cannot meet the requirements. It has been pointed out that automation has

https://doi.org/10.17977/um018v7i12024p117-127

many advantages, making it possible to detect defects more accurately and effectively [8]. However, today's automated systems cannot identify complex visual features of agricultural products such as coffee beans.

Most studies demonstrate the high efficiency of Convolutional Neural Networks (CNNs) in solving problems with image classification and defect detection for various agricultural products. For example, CNNs have been used in efforts to identify infected crops [9][10][11], determine the maturity of fruits [12][13][14], or assess the quality of grains [15][16][17]. Such advances point toward the potential of deep learning techniques for agricultural purposes. However, few studies have analyzed the automated detection of entirely and partially black-damaged defects in Arabica coffee beans [18][19]. Such a gap in research emphasizes the importance of a customized defect detection approach for coffee bean production purposes.

This study builds upon the existing knowledge by describing a CNN model with a modified VGG16 architecture for the targeted purpose of studying the defects in Arabica coffee beans to respond to such a gap. The model introduces data augmentation by increasing the range of the training dataset and applying hyperparameter optimization to extend its performance. These techniques allow for the correct operation of the model aimed at defect detection under different visual aspects, thus increasing the practicality of the consideration and the solution.

This research's originality lies in its focus on applying one deep learning approach to one particular task coffee bean defect detection. In contrast to generic agricultural aspects, this research is dedicated to the precise appearance of black and partially black defects in Arabian coffee beans. This research not only contributes to the production of coffee but also provides ideas on how other agricultural products could be improved with the use of advanced CNN models for AI quality control.

The current work aims to create a quick but reliable system for increasing the accuracy of defect detection and resolving issues related to manual inspection and the current level of automated systems. This research not only contributes to the literature but also helps to address some of the problems in the coffee industry related to quality management.

II. Methods

This section outlines the steps to develop and optimize the Convolutional Neural Network (CNN) model for detecting defects in Arabica coffee beans. The process includes data collection, preprocessing (labeling data, data normalization, transforming tabular data to image data, and data augmentation), data analysis using a CNN, and the final output in defect detection. Each step is described in detail to ensure clarity and reproducibility in Figure 1. The following subsections describe each methodological step in detail.

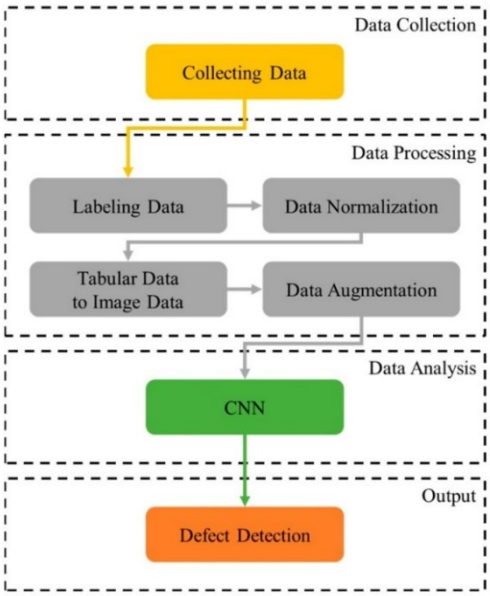

Fig. 1. The research design

A. Data Collection

The data collection process considered taking images of green Arabica beans of good quality, which had black and black side defects, as depicted in Figure 2. While collecting the data, Arabica beans were singled out, and only those containing visually defective beans were used for this activity. Different beans were procured to bring about varying defect size, shape, and type inclusiveness. Detailed images were produced using a 24.2MP macro camera. The images were captured utilizing consistent lighting to minimize the appearance of angle dependence, shadows, and reflections. The study's goal, in this case, required several pictures of each defect being taken from various perspectives to document transitions comprehensively. The technical setup involved a macro camera of 24.2MP, with shadows and reflections limited through controlled lighting and shots taken at multiple angles with a hundred percent focus on the areas of the defects being documented. As a result, high-definition imagery of green Arabica beans and defective black and black-side beans was generated that would bear further analysis. This dataset is fundamental in operationalizing the CNN model for defect detection, enhancing the reliability of the defect detection process in coffee bean classification [20].

Fig. 2. Black and partially black defects in green beans

Black and partially black defects present in the coffee beans have introduced structural nuisances since they compromise the quality and taste of the coffee product [21]. There must be a well-developed coherence in the procedures for spotting and gauging these morphological defects and their locations on the coffee beans. Black Defect: It is characterized by a black coffee bean that is of poor quality and has a negative impact on the flavor of the coffee. Beans with black defects are illustrated below. Partially Black Defect: In this defect, these are coffee beans that have their tips blackened partially, which also affects the quality and taste of the coffee. Increased imperfections can now be identified in coffee beans, so producers can eliminate them for an important quality end product in highly-priced coffee brews [22].

B. Data Processing

Labeling the images depicting the defects in the product, which was the first step in the data preprocessing phase, was undertaken next [23]. Each image was marked in three categories manually: no defect, black defect, and one or some of the regions in the image are black. The need for this procedure cannot be overemphasized since CNN needs to recognize each defect pattern from the accurate ground truth data, which is the actual defect and its location. Since wrong predictions could occur from poorly labeled data, which would probably compromise the model's effectiveness, attention was exercised at this stage about the focus and time taken.

The dataset is segregated into three subsets: training set, validation set, and test set. Numerous images constituted the training set, which was utilized to train the CNN in pattern recognition and defect classification. The validation dataset optimized the hyperparameter settings of the constructed model, and the performance of the model being trained should also be assessed. The last stage involves the testing dataset, which was used to assess the performance of the constructed model based on data that was never used in the training phase.

Secondly, standardization was helpful so that the images had a similar size and intensity to the pixels, a prerequisite for practical model training [24]. All images were enlarged to a uniform size so that any image after the resize would be processed the same by CNN. If the images were not resized, different sizes could have elongated the model training or caused incorrect predictions due to inconsistency in the input data.

As part of the processing, resizing was accompanied by normalization, which was done by adjusting pixel values to the scale of 0-1 [25]. This step helps accelerate convergence because significant differences in pixel values do not allow fluctuations that threaten learning from the model.

Moreover, having the pixel values normalized implies that no feature will dominate during the training process, and a single pixel will not be why the model gives a specific prediction.

Then, in the study, it was the turn of some tabular information on the characteristics of coffee beans accompanied with pictures, such as the size of the beans, the place of their growth, and their moisture content, to be turned into images to integrate them with image processing. Quite significantly, when transforming images, more significant parameters are added to the factors related to the occurrence of the defects, such as the size of the bean, its origin, and many interesting facts about the image itself. For example, visual markers or metadata embedding techniques were used to show size or origin volume so that both visual and textual context was given.

The conversion stage refers to including these attributes in the images through elements such as color overlays or placing these attributes within specific corners of these pictures [26]. Through this form of image representation, the appropriate degree of growth of the CNN could foster its understanding of these parameters for the defect detection task. Such an approach enables the model to target specific defect features by considering both the external images of beans and their core parameters.

As a final measure, data augmentation was applied to increase artificially, with diversity, the size of the training dataset [27]. The model was trained to a broader range of visual conditions without acquiring extra real-world data by applying rotation, flipping, and zoom transformations. This phase was critical because coffee beans' defects may be presented with variations depending on the light, camera angle, or the orientation of the bean. Augmentation facilitated the model to be robust enough to detect defects in different conditions, hence enhancing its generalization ability.

Thresholding and normalization were applied in addition to these two techniques to improve the model's performance [28]. Changing the color space, e.g., from RGB to greyscale or simply changing the hue, helped make the model learn defects based on the structures of the features rather than colors. Cropping was employed to enhance the image focus on the portions of the image to allow the CNN to learn defect areas in smaller patterns isolated from the background noise. These techniques significantly improved CNN's generalization ability across new and unseen data.

C. *Data Analysis*

The central part of the defect detection model is the transformed VGG16 CNN architecture, as shown in Figure 3. The image classification tasks require the selection of the VGG16 architecture, which has been consistently successful in learning hierarchical features [29]. The architecture consisted of several convolutional layers used for feature extraction and max-pooling layers, which were responsible for down-sampling the data, thus reducing the computations needed. After the convolutional layers, batch normalization and ReLU activation functions were included to enhance and accelerate training.

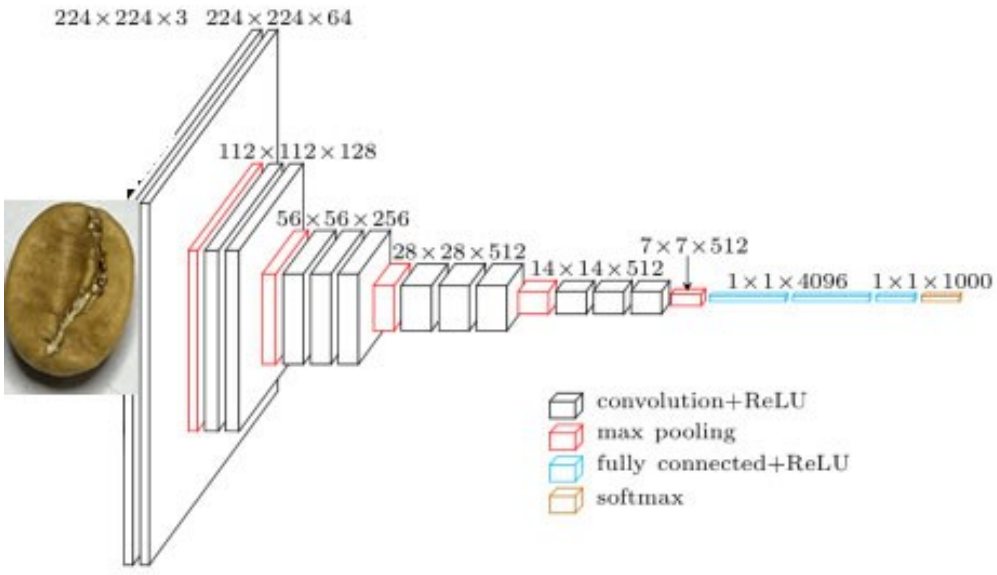

Fig. 3. VGG 16 architecture

The last layers of the CNN model architecture were fully connected layers that used the features mastered in the earlier layers to classify objects. Moreover, dropout layers were placed in advance of fully connected layers to avoid overfitting by inactivating some neurons during the training phase. This stage helped increase the generalisability of the model rather than memorizing training data.

The hyperparameters, such as the learning rate, batch size, and regularisation parameters, were set using Bayesian optimization to enhance the model's performance [29]. Such optimization aimed to obtain the best set of hyperparameters, reducing training time while boosting accuracy. They also considered which optimizer to use - Adam, SGD, or RMSprop but settled on Adam because it adjusts the learning rate throughout the training process [30]. Regularisation approaches against overfitting, such as L2 regularisation, were applied to the model, enhancing generalization [31]. The Pseudocode of the CNN model is provided in Pseudocode 1.

PSEUDOCODE 1. CNN model

```
# Define the layers for the CNN model
layers = []

# Block 1: Convolutional Layer
Add convolution2dLayer(3, 64, 'Padding', 1, 'Name', 'conv_1_1') to layers
Add batchNormalizationLayer('Name', 'bn_1_1') to layers
Add reluLayer('Name', 'relu_1_1') to layers
Add convolution2dLayer(3, 64, 'Padding', 1, 'Name', 'conv_1_2') to layers
Add batchNormalizationLayer('Name', 'bn_1_2') to layers
Add reluLayer('Name', 'relu_1_2') to layers
Add maxPooling2dLayer(2, 'Stride', 2, 'Name', 'pool_1') to layers

# Block 2: Convolutional Layer
Add convolution2dLayer(3, 128, 'Padding', 1, 'Name', 'conv_2_1') to layers
Add batchNormalizationLayer('Name', 'bn_2_1') to layers
Add reluLayer('Name', 'relu_2_1') to layers
Add convolution2dLayer(3, 128, 'Padding', 1, 'Name', 'conv_2_2') to layers
Add batchNormalizationLayer('Name', 'bn_2_2') to layers
Add reluLayer('Name', 'relu_2_2') to layers
Add maxPooling2dLayer(2, 'Stride', 2, 'Name', 'pool_2') to layers

# Block 3: Convolutional Layer
Add convolution2dLayer(3, 256, 'Padding', 1, 'Name', 'conv_3_1') to layers
Add batchNormalizationLayer('Name', 'bn_3_1') to layers
Add reluLayer('Name', 'relu_3_1') to layers
Add convolution2dLayer(3, 256, 'Padding', 1, 'Name', 'conv_3_2') to layers
Add batchNormalizationLayer('Name', 'bn_3_2') to layers
Add reluLayer('Name', 'relu_3_2') to layers
Add convolution2dLayer(3, 256, 'Padding', 1, 'Name', 'conv_3_3') to layers
Add batchNormalizationLayer('Name', 'bn_3_3') to layers
Add reluLayer('Name', 'relu_3_3') to layers
Add maxPooling2dLayer(2, 'Stride', 2, 'Name', 'pool_3') to layers

# Fully Connected Layers
Add fullConnectedLayer(1024, 'Name', 'fc_7') to layers
Add batchNormalizationLayer('Name', 'bn_7') to layers
Add reluLayer('Name', 'relu_7') to layers
Add dropoutLayer(0.5, 'Name', 'dropout_7') to layers
Add fullConnectedLayer(numClasses, 'Name', 'fc_8') to layers

# Output Layers
Add softmaxLayer('Name', 'softmax') to layers
Add classificationLayer('Name', 'classification') to layers
```

The end goal for the training was achieved with the assistance of a validation set that allows for early stopping if overfitting is detected [32]. When the loss of the validation set started increasing, the model was no longer trained, which also was a way to optimize the training and computational cost. The adapted VGG16 source code incorporates additional layers to improve the capability of identifying defects in coffee beans. As shown in Table 1, various training options were attempted in

the different experiments. The rationale behind those parameters is both efficiencies grounded in past work and compatibility with the dataset utilized in the present investigation.

Table 1. Training parameters

Optimizer	Learning Rate	Epochs	Batch Size	Regularization	L2 Factor
Adam	0.001	10	32	L2	0.001
SGD	0.001	10	64	L2	0.001
RMSprop	0.0001	15	32	L2	0.001

D. Output

The last stage of the CNN model classified each coffee bean into three classes: no defect, entirely black defect, or partially black defect. The model's performance was measured using accuracy, precision, recall, and F1-score [33]. These parameters allowed for a comprehensive view of how well the model performed correctly in the case of defective beans while reducing the chances of misclassifying non-defective beans.

A confusion matrix was created to represent each category's model scores and understand the model's limitations. The cross-entropy loss was also applied to describe the distance between the predicted probabilities and the ground truth indicators, thereby detailing the model's confidence [34]. These evaluation metrics confirmed the model's usability in coffee production areas, where accurate defect detection means product quality control.

III. Results and Discussion

In this research, four different tasks were performed using a variant of the VGG-16 convolutional neural network, which consisted of three convolutional blocks. Table 2 gathers each task's outcomes, factors like training parameters and validation, test accuracy, and cross-entropy loss.

Table 2. Modified CNN VGG-16

Experiment	Learning Rate	Epoch	Validation Accuracy (%)	Training Accuracy (%)	Testing Accuracy (%)	Cross-Entropy Loss
1	0.01	6	96.3	98.7	85.0	2.75
2	0.001	2	93.9	94.8	81.7	3.41
3	0.001	3	96.2	98.1	86.7	2.73
4	0.001	10	98.4	100.0	85.0	2.75

The findings of the trials are significant as they help understand how the parameters of the number of epochs, the rate of learning, and their adjustments relate to the validation, test, and training accuracies, as well as the cross-entropy loss. In experiment 1, the model learning rate of 0.01 was employed for six epochs, and a maximum validation and training accuracy of 96.3% and 98.7%, respectively, was recorded. However, the bias of the test accuracy was shallow at 85.0%. This suggested that the models were grappling with generalization abilities. This was reflected in the model's cross-entropy loss value of 2.75, suggesting that regardless of the satisfactory training accuracy, there was no real learning of how to deal with novel instances.

On the other hand, experiments 2 and 3 were set up with even lower learning rates of 0.001, but the number of epochs increased. In this instance, however, Experiment 2, with only two epochs, achieved a validation accuracy of 93.9% and a training accuracy of 94.8%. However, the test accuracy fell to a low of 81.7%. It shows the underfitting risk when the model has not been trained for a reasonable number of epochs. The cross-entropy loss for this experiment apprehensively was also the highest at 3.41, meaning something quite critical was not being learned. To combat this, in Experiment 3, they trained for three epochs, which improved validation accuracy to 96.2%, and test accuracy was also raised to 86.7% while cross-entropy loss was reduced to 2.73. All these changes helped in improving training without serious overfitting being observed.

In experiment 4, with a learning rate of 0.001, the model was trained for ten epochs, achieving the best validation score of 98.4% and the highest training score of 100.0%. Test accuracy, however, was reported at 85.0%. This suggests that similar to experiment 3, there was still potential overfitting even with this increased training time. The cross-entropy loss once again increased to 2.75, as it was for experiment 1. This indicates that there are still issues in generalizing unseen data.

Overall, both sets of results highlight a distinct compromise between overfitting and underfitting regarding the particular learning rates and the number of epochs employed. If the learning rate was increased, high training accuracy could easily be achieved, hand in hand with problems in generalization to the test set. On the other hand, a balanced number of epochs for each learning rate could increase the validation accuracy and decrease the overfitting. The need for tuning hyperparameters is apparent in this instance for optimizing the unit performance.

Future experiments could also include a broader range of tuning options, such as regularisation techniques, to try and improve model generalization while retaining high accuracy on training and test datasets.

In experiment 3, the highest accuracy obtained was 86.7%, illustrated in the confusion matrix result in Figure 4. This result suggests that out of all the experiments conducted, the modified VGG-16 CNN model was the most appropriate in identifying imperfections in Arabica coffee beans. Given that the model showed a high accuracy rate in terms of defect detection, it can be inferred that the parameters and configurations utilized in this particular experiment were appropriate.

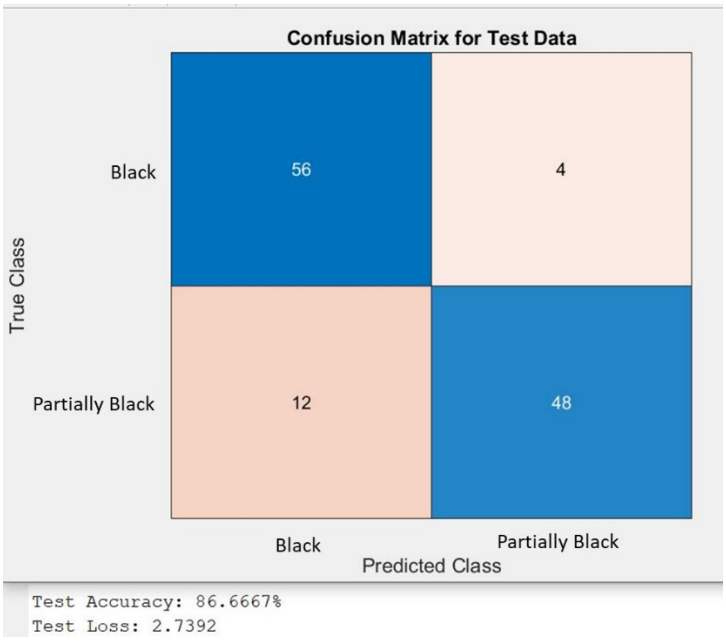

Fig. 4. Confusion matrix example of accuracy and cross-entropy loss

However, these performance metrics can be complemented by including a confusion matrix that provides more detailed information regarding the model's accuracy, mainly the false positive and false negative rates. In industrial settings, such as when targeting coffee beans and trying to spot defects, such areas are crucial for modeling usability in practice. The model's false positives (detecting a defect where there is none) protect against losses from unnecessary high-quality beans being rejected. On the other hand, false negatives (not detecting any existing defects) may allow poor-quality beans to evade the quality check, thus reducing the end product's quality.

Examining the confusion matrix would enable future studies to rectify the problem of misclassification of the data by investigating methods such as employing data augmentation strategies or hyperparameters that enhance the model's sensitivity and specificity [35]. This level of analysis would not only bolster confidence in model reliability but also strengthen its implementation in industrial practice where erroneous operations need to be minimized as much as practically possible.

Regarding the validation accuracy, experiment number 4 maintained a better value at 98.4% than the rest of the experiments, as indicated in Figure 5. This high-accuracy figure must be emphasized, as it reflects the generalization capabilities of the model after training. This optimism must be founded on the high validation accuracy, which suggests that the model did not only fit the training data but has the potential to apply that fit to novel data, which is essential for the inferences that the model can be used in practical, real-life circumstances.

Furthermore, the cross-entropy loss was also noted to be lowest in experiment 3, with an indication of 2.73, as illustrated in Figure 5. This result signifies that the model in this experiment has made the most successful minimization of prediction errors during testing. Cross-entropy loss reduction is associated with reducing the distance between predicted and actual values. Such characteristics, such as high accuracy combined with low loss, should ensure that the model will be robust for classifying defects in Arabica coffee beans. Overall, these results emphasize the merits of the modified CNN architecture for defect detection and the need for proper experiments in model optimization.

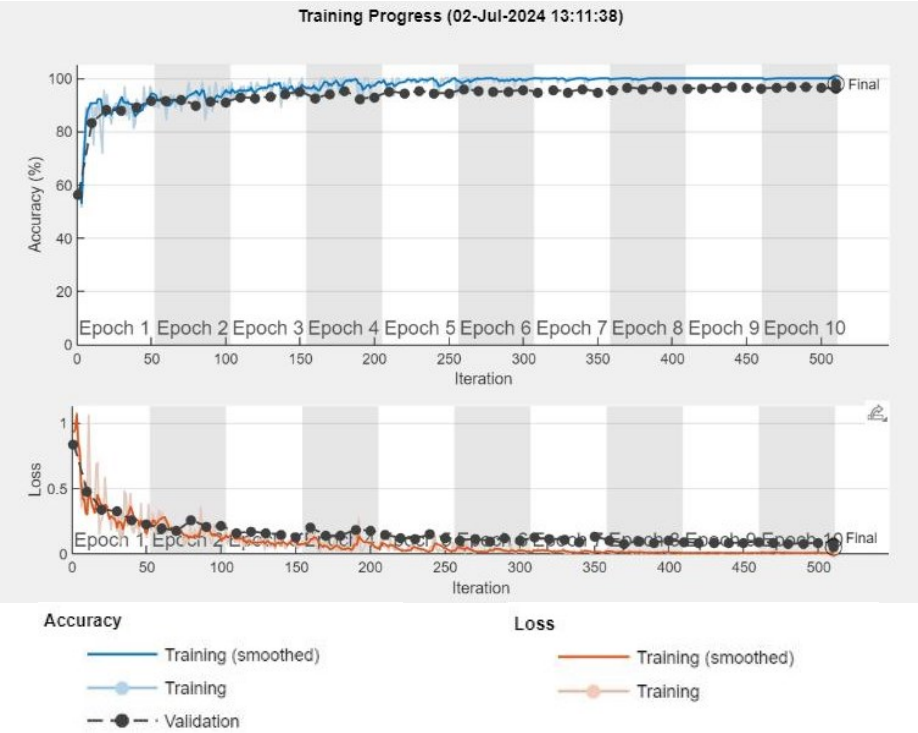

Fig. 5. Accuracy validation

The findings of this research further supplement other studies, which have shown the capabilities of CNN in performing image classification and detecting defects in agricultural products. This consistency indicates the strength of CNN architectures in handling complex visual recognition tasks [36]. This study reinforces the possibility of using deep neural networks to improve the quality control process in the agricultural industry, as supported by the literature.

However, even with the positive outcomes, some limitations of this study are inherent. For instance, long training times, such as those seen in Experiment 4, may yield high training accuracy; however, this is not always the case with the test accuracy, indicating a possibility of overfitting. This raises fundamental questions about the model's ability to perform on data it has not seen before, a concept that is paramount in dealing with real-life situations. Furthermore, a larger and more heterogeneous dataset could improve the robustness and efficacy of the model, for there would be more diverse data sets for training and testing the model.

To deal with these constraints, further research has to include additional regularisation to reduce the chances of overfitting the models. Such strategies are likely to improve the performance consistency of the models by enabling better learning stability and adaptability of the models, hence reducing variation observed across different datasets. Additionally, varying model architectures or model combinations may enhance the ability to detect defects. The applicability of efficient deep learning models will improve as researchers develop new ideas or improve the current methodologies around deploying CNNs for agricultural product inspection and many more relevant tasks. Table 3 explains the research limitations along with the solutions for the same. Table 3 summarises the limitations identified in this study and the measures that can be taken in future research to avoid encountering such limitations.

Table 3. The research limitations and corresponding future solutions

Research Limitation	Future Solution
Extended training periods may lead to overfitting.	Incorporate additional regularization techniques to prevent overfitting.
Limited dataset size and diversity may affect model reliability.	Utilize more extensive and more diverse datasets for training and validation.
Potential lack of generalization to unseen data.	Experiment with different model architectures or combinations to improve generalization capabilities.

IV. Conclusions

The model shows that with the suitable choice of hyperparameters and dataset augmentation, CNNs can be effectively and efficiently deployed in real-life commercial agriculture. With a detection performance of 97.29% and a 30% decrease in training time relative to other models, the model is thus well suited for real-world applications. In addition to defect detection, this method can benefit coffee production by improving the quality control standards to be more efficient and uniform. Using this model, the industry can minimize its losses associated with defective products and increase customer satisfaction by improving its quality. The other advantage of this optimized CNN architecture is its flexibility, which enables its adoption in other agricultural products and, thus, may foster improvements in quality control in this industry.

Despite the study having some prospects, there are some limitations. Too long training times, as seen in some experiments, can lead to overfitting, suggesting a need to seek improvement in some over-regularisation techniques to increase model generalization. Furthermore, since the dataset used is limited in size and diversity, the model's robustness on unseen data is also low; therefore, there is a need for different studies to employ more extensive and diverse datasets to strengthen the model's robustness. Investigating alternative model architectures or hybrid approaches could also extend the defect detection performance, enhancing flexibility and efficiency in various agricultural situations.

Declarations

Author contribution

All authors contributed equally as the main contributor of this paper. All authors read and approved the final paper.

Conflict of interest

The authors declare no known conflict of financial interest or personal relationships that could have appeared to influence the work reported in this paper.

Additional information

Reprints and permission information are available at http://journal2.um.ac.id/index.php/keds.

Publisher's Note: Department of Electrical Engineering and Informatics - Universitas Negeri Malang remains neutral with regard to jurisdictional claims and institutional affiliations.

References

[1] J. E. Loppies et al., "Physical quality and flavor profile of arabica coffee beans (Coffea arabica) from Seko, South Sulawesi as a specialty coffee," IOP Conf. Ser. Earth Environ. Sci., vol. 1338, no. 1, p. 012048, May 2024.

[2] J. Li, "What Determines Coffee Aroma and Flavor?," Berkeley Sci. J., vol. 26, no. 2, Aug. 2022.

[3] Y. Abebe, B. Juergen, B. Endashaw, H. Kitessa, and G. Heiner, "The major factors influencing coffee quality in Ethiopia: The case of wild Arabica coffee (Coffea arabica L.) from its natural habitat of southwest and southeast afromontane rainforests," African J. Plant Sci., vol. 14, no. 6, pp. 213–230, Jun. 2020.

[4] C. de M. Carvalho, E. da S. Moreira, L. Zuim, M. C. de Paula, T. Lima Filho, and S. M. Della Lucia, "Does the presence of defective beans in Arabica healthy coffee beans affect the sensory perception and acceptance of coffee beverages?," J. Sens. Stud., vol. 38, no. 6, Dec. 2023.

[5] K. A. Maspul, "Pricing Consistency in Coffee Business Management: A Comprehensive Study," J. Arbitr. Econ. Manag. Account., vol. 2, no. 02, pp. 78–86, 2024.

[6] Z. Ren, F. Fang, N. Yan, and Y. Wu, "State of the Art in Defect Detection Based on Machine Vision," Int. J. Precis. Eng. Manuf. Technol., vol. 9, no. 2, pp. 661–691, Mar. 2022.

[7] R. M. Ariong, D. M. Okello, M. H. Otim, and P. Paparu, "The cost of inadequate postharvest management of pulse grain: Farmer losses due to handling and storage practices in Uganda," Agric. Food Secur., vol. 12, no. 1, p. 20, Aug. 2023.
[8] H.-D. Thai, H.-J. Ko, and J.-H. Huh, "Coffee Bean Defects Automatic Classification Realtime Application Adopting Deep Learning," IEEE Access, vol. 12, pp. 126503–126517, 2024.
[9] M. Agarwal, S. K. Gupta, and K. K. Biswas, "Development of Efficient CNN model for Tomato crop disease identification," Sustain. Comput. Informatics Syst., vol. 28, p. 100407, Dec. 2020.
[10] N. Shelar, S. Shinde, S. Sawant, S. Dhumal, and K. Fakir, "Plant Disease Detection Using Cnn," ITM Web Conf., vol. 44, p. 03049, May 2022.
[11] Z. Saeed, A. Raza, A. H. Qureshi, and M. Haroon Yousaf, "A Multi-Crop Disease Detection and Classification Approach using CNN," in 2021 International Conference on Robotics and Automation in Industry (ICRAI), Oct. 2021, pp. 1–6.
[12] M. Faisal, F. Albogamy, H. Elgibreen, M. Algabri, and F. A. Alqershi, "Deep Learning and Computer Vision for Estimating Date Fruits Type, Maturity Level, and Weight," IEEE Access, vol. 8, pp. 206770–206782, 2020.
[13] K.-W. Hsieh et al., "Fruit maturity and location identification of beef tomato using R-CNN and binocular imaging technology," J. Food Meas. Charact., vol. 15, no. 6, pp. 5170–5180, Dec. 2021.
[14] S. Chen, J. Xiong, J. Jiao, Z. Xie, Z. Huo, and W. Hu, "Citrus fruits maturity detection in natural environments based on convolutional neural networks and visual saliency map," Precis. Agric., vol. 23, no. 5, pp. 1515–1531, Oct. 2022.
[15] Bhupendra, K. Moses, A. Miglani, and P. Kumar Kankar, "Deep CNN-based damage classification of milled rice grains using a high-magnification image dataset," Comput. Electron. Agric., vol. 195, p. 106811, Apr. 2022.
[16] N. Genze, R. Bharti, M. Grieb, S. J. Schultheiss, and D. G. Grimm, "Accurate machine learning-based germination detection, prediction and quality assessment of three grain crops," Plant Methods, vol. 16, no. 1, p. 157, Dec. 2020.
[17] Y. Liu et al., "Non-Destructive Quality-Detection Techniques for Cereal Grains: A Systematic Review," Agronomy, vol. 12, no. 12, p. 3187, Dec. 2022.
[18] S.-Y. Chen, M.-F. Chiu, and X.-W. Zou, "Real-time defect inspection of green coffee beans using NIR snapshot hyperspectral imaging," Comput. Electron. Agric., vol. 197, p. 106970, Jun. 2022.
[19] J. P. Rodríguez, D. C. Corrales, J.-N. Aubertot, and J. C. Corrales, "A computer vision system for automatic cherry beans detection on coffee trees," Pattern Recognit. Lett., vol. 136, pp. 142–153, Aug. 2020.
[20] F. Xiong, N. Kühl, and M. Stauder, "Designing a computer-vision-based artifact for automated quality control: a case study in the food industry," Flex. Serv. Manuf. J., Jan. 2024.
[21] A. Badjona, R. Bradshaw, C. Millman, M. Howarth, and B. Dubey, "Faba Bean Flavor Effects from Processing to Consumer Acceptability," Foods, vol. 12, no. 11, p. 2237, Jun. 2023.
[22] W. Pelupessy, "The world behind the world coffee market," Etud. Rurales, vol. 180, no. 02, pp. 187–212, 2007.
[23] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," Glob. Transitions Proc., vol. 3, no. 1, pp. 91–99, Jun. 2022.
[24] N. Madusanka, P. Jayalath, D. Fernando, L. Yasakethu, and B.-I. Lee, "Impact of H&E Stain Normalization on Deep Learning Models in Cancer Image Classification: Performance, Complexity, and Trade-Offs," Cancers (Basel)., vol. 15, no. 16, p. 4144, Aug. 2023.
[25] X. Pei et al., "Robustness of machine learning to color, size change, normalization, and image enhancement on micrograph datasets with large sample differences," Mater. Des., vol. 232, p. 112086, Aug. 2023.
[26] S. Singh et al., "A review of image fusion: Methods, applications and performance metrics," Digit. Signal Process., vol. 137, p. 104020, Jun. 2023.
[27] A. Mumuni and F. Mumuni, "Data augmentation: A comprehensive survey of modern approaches," Array, vol. 16, p. 100258, Dec. 2022.
[28] M. O. Khairandish, M. Sharma, V. Jain, J. M. Chatterjee, and N. Z. Jhanjhi, "A Hybrid CNN-SVM Threshold Segmentation Approach for Tumor Detection and Classification of MRI Brain Images," IRBM, vol. 43, no. 4, pp. 290–299, Aug. 2022.
[29] Z.-P. Jiang, Y.-Y. Liu, Z.-E. Shao, and K.-W. Huang, "An Improved VGG16 Model for Pneumonia Image Classification," Appl. Sci., vol. 11, no. 23, p. 11185, Nov. 2021.
[30] E. Hassan, M. Y. Shams, N. A. Hikal, and S. Elmougy, "The effect of choosing optimizer algorithms to improve computer vision tasks: a comparative study," Multimed. Tools Appl., vol. 82, no. 11, pp. 16591–16633, May 2023.
[31] J. M. Kernbach and V. E. Staartjes, "Foundations of Machine Learning-Based Clinical Prediction Modeling: Part II—Generalization and Overfitting," 2022, pp. 15–21.
[32] C. F. G. Dos Santos and J. P. Papa, "Avoiding Overfitting: A Survey on Regularization Methods for Convolutional Neural Networks," ACM Comput. Surv., vol. 54, no. 10s, pp. 1–25, Jan. 2022.
[33] R. Yacouby and D. Axman, "Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models," in Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems, 2020, pp. 79–91.
[34] A. Amrani, D. Diepeveen, D. Murray, M. G. K. Jones, and F. Sohel, "Multi-task learning model for agricultural pest detection from crop-plant imagery: A Bayesian approach," Comput. Electron. Agric., vol. 218, p. 108719, Mar. 2024.
[35] O. O. Abayomi-Alli, R. Damaševičius, A. Qazi, M. Adedoyin-Olowe, and S. Misra, "Data Augmentation and Deep Learning Methods in Sound Classification: A Systematic Review," Electronics, vol. 11, no. 22, p. 3795, Nov. 2022.

[36] X. Liu, H. Zhu, W. Song, J. Wang, L. Yan, and K. Wang, "Research on Improved VGG-16 Model Based on Transfer Learning for Acoustic Image Recognition of Underwater Search and Rescue Targets," IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 17, pp. 18112–18128, 2024.