Enhancing green coffee quality assessment through deep learning

Reinaldo Gonçalves Pereira Neto*, Pedro Moisés de Sousa*, Larissa Ferreira Rodrigues Moreira*,
Pedro Ivo Vieira Good God† and João Fernando Mari*
*Instituto de Ciências Exatas e Tecnológicas
Universidade Federal de Viçosa - UFV, Rio Paranaíba, MG, Brazil
Email: {reinaldo.neto, pedromoises, larissa.f.rodrigues, joaof.mari}@ufv.br
†Instituto de Ciências Agrárias
Universidade Federal de Viçosa - UFV, Rio Paranaíba, MG, Brazil
Email: pedro.god@ufv.br

***Abstract*—Coffee is the world's most consumed commodity beverage, vital for the Brazilian market. Assessing coffee bean quality through visual features is essential for market value. However, human-based visual analysis has limitations. Deep neural networks, particularly CNNs, offer a promising solution by automating this process. In this work, we propose an evaluation of deep learning models and training strategies to classify green coffee beans automatically. We evaluate four CNN architectures: AlexNet, ResNet-50, MobileNet V3, and EfficientNet B4. After a hyperparameter optimization step, the models were fine-tuned, and we evaluated the impact of data augmentation strategies on the classification performance through the USK-Coffee dataset. EfficientNet B4 excels, achieving 0.8844 accuracy when trained with data augmentation. Our findings showcase deep learning's potential for coffee quality assessment, aiding professionals in classifying and guaranteeing coffee quality and value.**

***Index Terms*—green coffee, coffee bean, deep learning, classification, data augmentation, optimization.**

I. Introduction

Coffee is the most consumed commodity beverage worldwide, and the demand for high-quality coffee is growing [1]. Brazil is the largest producer and exporter of green coffee beans and the second largest consumer of the beverage in the world, making coffee one of the main commodities in the Brazilian market [2]. The commercialization of coffee is a significant contributor to the income of many emerging economies, and characteristics such as color, morphology, shape, and size can be used to define the quality and market value of a grain [1], [3].

Visual analysis of coffee beans, as well as other processes that depend on visual analysis performed by human beings, is subject to errors owing to the inherent subjectivity of the problem and the fatigue imposed by the repetitive process. One should also consider the problem of scale because there is a limit on the samples that a human expert can analyze in a time interval.

Deep neural networks such as Convolutional Neural Network (CNN)s allow the hierarchical extraction of features from images, establishing classification models that match or surpass the accuracy of human experts in many tasks [4]. After the models are effectively trained, they can be used to classify hundreds of samples at intervals of a few minutes. Its training is complex, as it requires a large number of images for training as well as adequate adjustments of hyperparameters, regularization techniques, data augmentation procedures, and transfer learning. These characteristics must be analyzed through experiments to arrive at suitable models and training strategies for each problem [5], [6].

This paper proposes a method for automatically classifying coffee beans to enhance green coffee quality using Deep Learning techniques. This work aims to analyze the performance of CNN models to classify images of green coffee beans. We analyzed AlexNet, ResNet-50, MobileNet V3, and EfficientNet B4 architectures. We performed a grid-search-based hyperparameter optimization and compared the impact of a data augmentation strategy on the prediction performance. Moreover, as far as we know, our result is the best obtained for coffee bean classification using the dataset evaluated in this study. We believe that our proposed method can contribute to future research intended to help assess the final drink's quality, generating value for the product.

The remainder of this paper is organized as follows. Section II presents related work. In Section III, we describe the material and experiment design. Results are presented in Section IV. In Section V, we expose conclusions and future work.

II. Related Work

Recently, several studies have been dedicated to evaluating the quality of coffee production using Computer Vision techniques. Oliveira et al. [3] proposed a method for quantifying the quality of green coffee beans by classifying images of the beans using neural networks and a Bayes classifier based on color characteristics obtained from the CIE L*a*b*. Garcia et al. [1] employed the K-Nearest Neighbor method to evaluate the quality of coffee beans and identify their defects. They extracted different features from the images of the beans, including area, circularity, damaged-area ratio, and eccentricity. This method classifies coffee beans based on their quality into four categories: very low, low, high, and very high.

Additionally, it categorizes them according to their defects as normal, black, sour, broken, very long berry, and small.

Costa et al. [7] obtained colorimetric variables using the RGB, CIE L*a*b*, and HSV models combined with the K-means technique to classify coffee fruits. They reduced the colorimetric variables using Principal Component Analysis. However, this approach requires a greater computational load and increases processing time.

Pradana-López et al. [8] applied a ResNet-34 CNN to support the quality control and detection of adulteration of Arabica and Robusta coffee with other foods, such as chicory and barley. They considered ground coffee images and, owing to dataset size limitations, applied a transfer learning technique considering the weights from ImageNet.

Chang et al. [9] applied a CNN inspired by AlexNet to inspect defects in coffee beans. Five defects were classified: cut, immature, partialsour, slight insect damage, and withered. The method has demonstrated an improved generalization ability, but its classification performance must also be evaluated for distorted samples.

Wang et al. [10] proposed a model based on CNN to support the automatic detection of defective coffee beans to guarantee higher-quality coffees using knowledge distillation, a method of transferring learning from a more complex machine to a simpler one, to increase the lightness of the model.

Tamayo-Monsalve et al. [11] evaluated different CNN architectures to classify spectral images of coffee fruits in various stages of ripening. However, using spectral images for coffee fruit classification has challenges, such as resource-intensive storage and processing, the need for specialized equipment, and extensive computational resources for effective analysis.

Febriana et al. [12] proposed using ResNet-18 and MobileNetV2 to classify a large dataset of green coffee beans categorized into four classes. However, the authors did not explore different types of data augmentation or hyperparameter adjustment strategies to improve classification performance.

In contrast to the previously mentioned studies, our study adopts a novel approach to improve green coffee quality assessment through images and deep learning. Unlike previous studies that focused on coffee bean classification using multispectral imagery, traditional classifiers, or CNNs without any optimization strategy, our study utilized deep learning techniques combined with hyperparameter optimization and data augmentation strategies. In addition, our study produces superior classification performance, surpassing the results achieved by previous approaches.

III. Material and methods

A. Dataset

For the experiments, we used the USK-Coffee dataset [12][1]. The USK-Coffee comprises 8,000 images of green coffee beans distributed in four classes: peaberry, longberry, premium, and defect. The images have 256 $\times$ 256 pixels and are equally distributed among the classes, with 2,000 images for each. In Figure 1, we present four samples from each class, randomly selected from the training set.

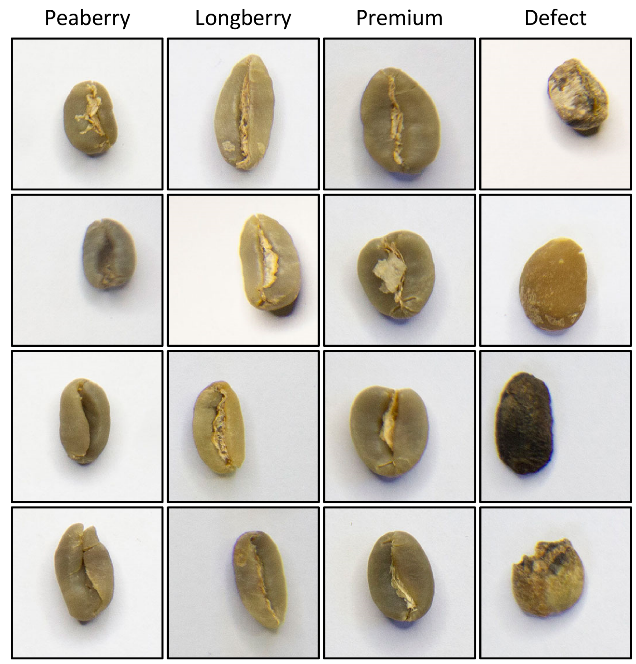

Fig. 1. Samples of each class in the USK-Coffee dataset.

B. Architectures

We evaluated four CNN architectures that were chosen because of their success in previous image classification tasks applied in agriculture [12] [13] [14].

AlexNet was proposed by Krizhevsky et al. [15] and was the first CNN that win the ILSVRC competition in 2012. AlexNet is composed of five convolutional layers with three polling layers intercalated and three fully connected layers before the output layer. Even with existing more modern CNN architectures, we choose to train the AlexNet because of its historical importance and as a baseline to compare with other architectures.

ResNet, or Residual Network, introduced by He et al. [16] addresses the gradient vanishing problem in deep networks through residual connections. Each residual block contains a sequence of convolutional layers, and skip connections enable skipping some residual blocks and feeding the next block's input with the previous one's output. ResNet-50 is a specific variant that consists of fifty convolutional layers.

MobileNet is a family of CNNs designed to run on mobile devices. The family starts with MobileNet V1 [17] with introduced deth-wise convolutions to reduce the number of parameters. MobileNet V2 [18] includes an inverted residual structure with residual connections between bottleneck layers. MobileNet V3, proposed by Howard et al. [19] extends MobileNet V2 with squeeze and excite operations in the residual layers.

EfficientNet is a groundbreaking CNN architecture introduced by Tan et al. [20]. It is designed to balance good image

[1] https://comvis.unsyiah.ac.id/usk-coffee/

classification performance with computational efficiency and achieves this through compound scaling, efficient building blocks, and global average pooling. EfficientNet offers variants from B0 to B7 to accommodate various hardware and performance needs. Its use of techniques like dropout normalization enhances its versatility. EfficientNet has consistently delivered state-of-the-art results in image classification, making it a popular choice in computer vision applications. For this work, we used the EfficientNet B4 variant.

C. Experiment design

The USK-Coffee dataset is provided with separate training, validation, and test sets. Then, we use the same dataset division to enable a fair comparison with the results reported in the literature.

Using Adam optimizer with cross-entropy loss, we fine-tuned models pre-trained with the ImageNet dataset [21] with all layers unfrozen. During the training, we decreased the learning rate after 10 epochs without the validation loss improvement. We early stopped the training process after 21 epochs without validation loss improvement (two complete cycles of learning rate decreasing without validation loss improvement).

We optimized the hyperparameter batch size (BS) and initial learning rate (LR) using a grid-search strategy with the search spaces $\{16, 32, 64, 128\}$ for BS and $\{0.01, 0.001, 0.0001, 0.00001\}$ for LR. No data augmentation was applied during the hyperparameter optimization step. The transformations applied to training, validation, and test sets were random resized crop and normalization using the ImageNet dataset statistics. After the hyperparameter optimization, all models were trained with and without data augmentation.

The data augmentation strategy applied consists of a random horizontal flip, followed by random rotation (-15° to 15°), random resized crop (patches between 80% and 100% the original size), color jittering (brightness, contrast, and saturation by a factor randomly chosen from 0.8 to 1.2), and random erasing (patches between 2% and 20% the original size). The images were also normalized by the mean and standard deviation of the ImageNet dataset. Image transformations for the validation and test sets are the same as those used in hyperparameter optimization.

D. Model evaluation

The validation accuracy was considered to select the best hyperparameter set during the hyperparameter optimization. To evaluate the final model, trained with and without data augmentation, besides accuracy, we used precision, recall, and F1-score for the validation and test sets. When comparing the indexes between the validation and testing sets, it is possible to access the capacity of the model to extrapolate the knowledge learned during training to unknown data.

Accuracy is computed as Equation (1):

$$Accuracy = \frac{TP + FP}{TP + FP + TN + FN} \tag{1}$$

Precision is calculated as Equation (2):

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

The recall is computed as Equation (3):

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

F1-Score is computed as Equation (4):

$$F1{-}Score = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{4}$$

where TP, TN, FP, and FN are true positive, true negative, false positive, and false negative samples, respectively.

E. Computational environment

The experiments were conducted in a PC equipped with a Core I5-4430 3.00 GHz CPU and 32 GB of RAM running Linux Ubuntu 20.04 LTS, equipped with a GPU NVIDIA GTX 1080 Ti. The experiments were developed using Python 3.9, PyTorch 2.0.1, torchvision 0.15.2 with CUDA Toolkit 10.1, Scikit-learn 1.2.2, and Matplotlib 3.7.1. The pre-trained models were obtained from torchvision.

IV. Results and Discussion

Table I shows the results of the hyperparameter optimization performed through grid search, as described in Section III-C. The table presents the set of parameters that resulted in the best validation accuracy for each model.

TABLE I
OPTIMIZED HYPERPARAMETER VALUES FOR EACH ARCHITECTURE.

Architecture	BS	LR	Acc. Val.	Epochs
AlexNet	16	0.00001	0.9169	13
ResNet-50	64	0.0001	0.9475	21
MobileNet V3	128	0.001	0.9606	50
EfficientNet B4	16	0.001	0.9469	24

Considering the hyperparameter obtained through the grid search optimization strategy and shown in Table I, we trained the models again with and without data augmentation operations. Figure 2 shows the evolution of the loss and accuracy curves during the training of these models for both training (pink line) and validation sets (green line). The vertical red line marks the epoch in which occurred the early stooping, i.e., we selected the model at this epoch to perform the predictions.

We visualized the accuracies through line charts to better understand the results presented in Table II. In Figures 3 and 4, it is possible to compare the performance among the trained models over the validation and test sets without and with data augmentation, respectively. Considering the test set, our best result when training the models without data augmentation was achieved by AlexNet and ResNet-50, with 0.8663 accuracy. When training the models using data augmentation, EfficientNet B4 resulted in the best test accuracy with 0.8844.

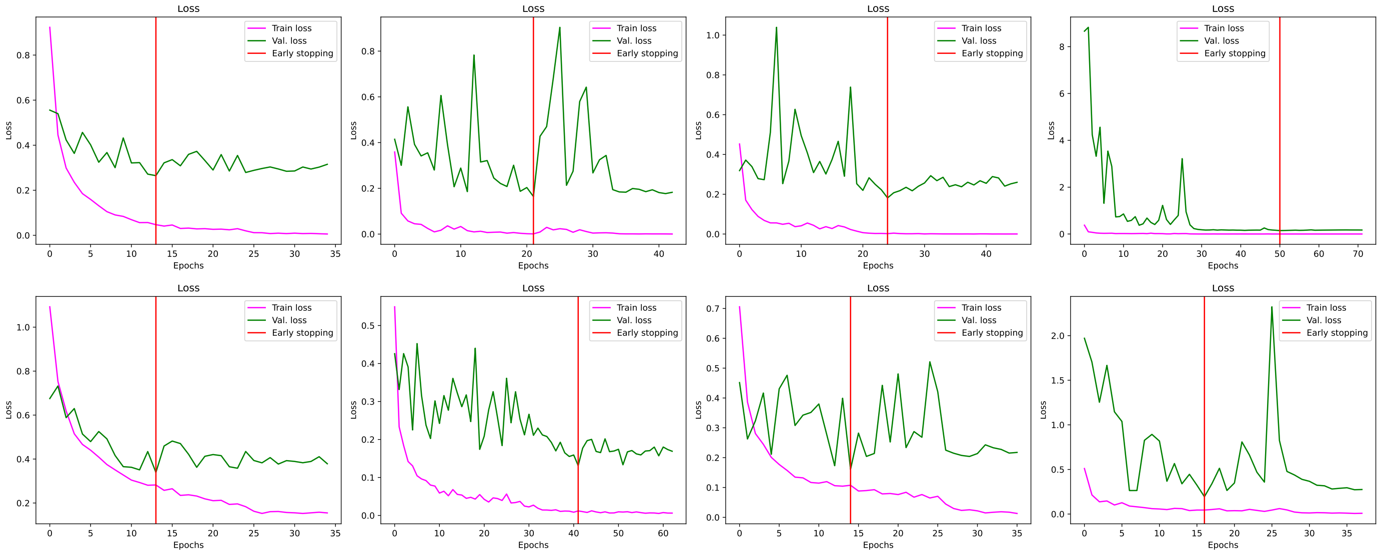

Fig. 2. Each column from left to right represents the models: AlexNet, ResNet-50, MobileNet V3, and EfficientNet b4. The first row shows the curves for training without data augmentation, and the second row shows the curves for training with a data augmentation strategy.

TABLE II
RESULTS OF THE EXPERIMENT WITH AND WITHOUT DATA AUGMENTATION.

		VAL				TEST				
Strategy	**Architecture**	**Acc.**	**Prec.**	**Rec.**	**F1**	**Acc.**	**Prec.**	**Rec.**	**F1**	**Epochs**
Without DA	**AlexNet**	0.9169	0.9172	0.9169	0.9165	0.8662	0.8660	0.8662	0.8650	13
	ResNet-50	0.9475	0.9478	0.9475	0.9476	0.8662	0.8680	0.8662	0.8667	21
	MobileNet V3	0.9606	0.9608	0.9606	0.9605	0.8506	0.8567	0.8506	0.8504	50
	EfficientNet b4	0.9469	0.9478	0.9469	0.9466	0.8588	0.8628	0.8588	0.8594	24
With DA	**AlexNet**	0.8788	0.8857	0.8787	0.8762	0.8581	0.8665	0.8581	0.8533	13
	ResNet-50	0.9644	0.9651	0.9644	0.9642	0.8694	0.8782	0.8694	0.8665	41
	MobileNet V3	0.9394	0.9425	0.9394	0.9391	0.8650	0.8705	0.8650	0.8620	16
	EfficientNet b4	0.9563	0.9580	0.9563	0.9561	0.8844	0.8899	0.8844	0.8827	30

Another interesting information we can observe in Figures 3 and 4 is the capability of each model to generalize what it learned during the training step to unsee data (test set). When trained without data augmentation, MobileNet V3 achieved an accuracy of 0.9606 in the validation set. Still, when we presented the test set to the model, it achieved only 0.8506, a difference of 0.11. But for AlexNet, the validation accuracy was 0.9169, and the test accuracy was 0.8663, a distance of 0.0506, indicating a better generalization capability.

Figure 5 shows a bar chart that compares the test accuracy of each model with and without data augmentation. In this figure, it is possible to observe that AlexNet cannot take advantage of data augmentation strategies. Still, all other models have their accuracies improved when trained with data augmentation. The best model for our problem was the EfficientNet B4 with a test accuracy of 0.8844 when trained with data augmentation, an increase of 0.0256 compared to training without data augmentation. It is interesting to observe that EfficientNet B4 trained without data augmentation (0.8588) is the second worst result above only MobileNet V3 (0.8506).

Finally, our findings indicate that the best result achieved

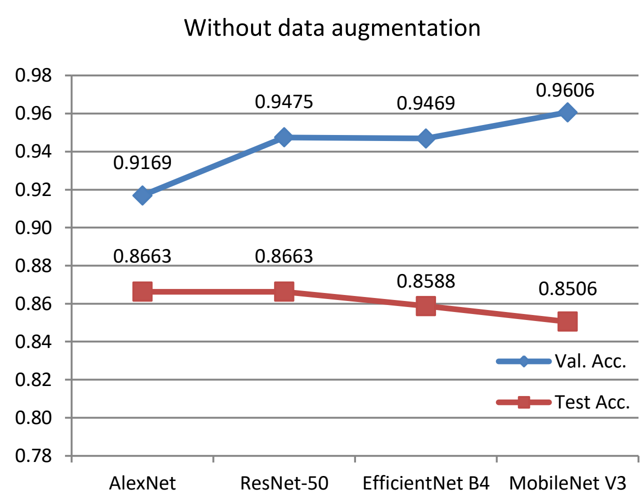

Fig. 3. Visualization of accuracies of validation and test sets when trained without data augmentation.

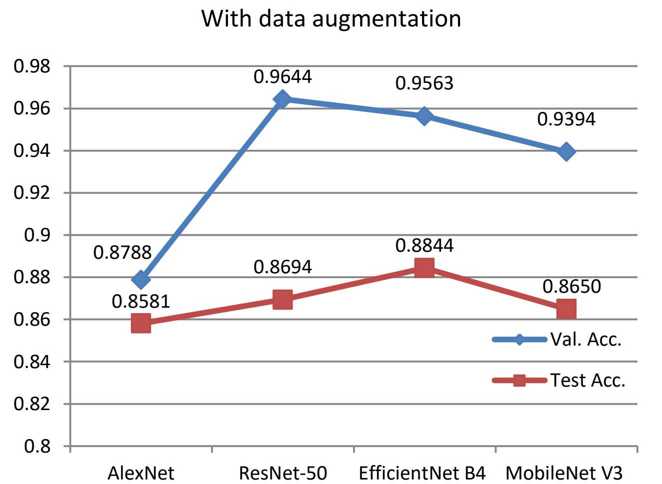

Fig. 4. Visualization of accuracies of validation and test sets when trained with data augmentation.

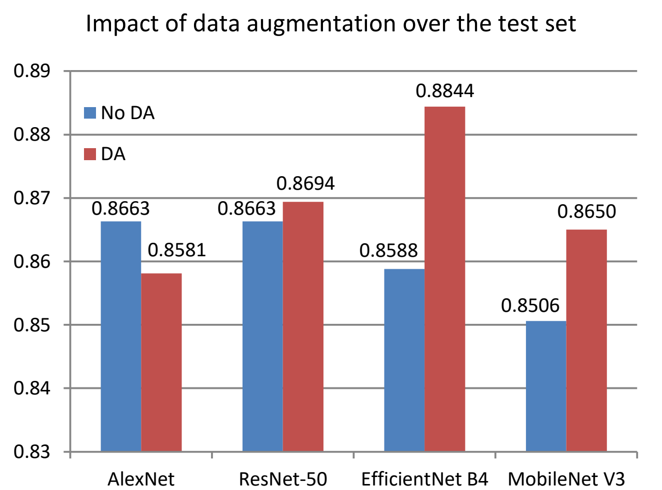

Fig. 5. Comparisson between models trained with and without data augmentation over the test set.

for the USK-Coffee dataset was 88.44% accuracy obtained with EfficientNet B4 with data augmentation, surpassing the performance of the previously published technique (88.31% in Febriana et al. [12]) which was the best until now.

V. Conclusions

In this work, we compared different deep-learning architectures to classify coffee green beans according to their quality based on digital images. We also compared the models' performance with and without data augmentation strategies. The models were fine-tuned over models pre-trained with ImageNet, and we performed a hyperparameter optimization step for each architecture.

AlexNet was the only model that did not take advantage of training with data augmentation strategies. The other models have improved results when trained with data augmentation. We highlight EfficientNet B4, in which the data augmentation is critical for good classification performance.

EfficientNet B4 achieved the best results over the test set (0.8844 accuracy) when considering the data augmentation procedure, but without performing data augmentation during the training phase, EfficientNet B4 was only the third best model, losing only MobileNet V3.

Our results demonstrate the viability of applying deep learning methods to assess the quality of coffee drinks by classifying green coffee bean images. This work paves the way to develop accessible and efficient applications to help professionals in classifying and selecting beans during the coffee production chain.

Future works include testing more deep learning architectures for image classification and other training strategies, such as more data augmentation transformations. Training and testing the models with other datasets to improve the generalization capability of the models.

Acknowledgments

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. R. G. Pereira Neto received a scholarship from PIBIC/FAPEMIG.

References

[1] M. García, J. E. Candelo-Becerra, and F. E. Hoyos, "Quality and Defect Inspection of Green Coffee Beans Using a Computer Vision System," *Applied Sciences*, vol. 9, no. 19, 2019.

[2] Y. M. Guimarães, J. H. P. P. Eustachio, W. Leal Filho, L. F. Martinez, M. R. do Valle, and A. C. F. Caldana, "Drivers and barriers in sustainable supply chains: The case of the Brazilian coffee industry," *Sustainable Production and Consumption*, vol. 34, pp. 42–54, 2022.

[3] E. M. de Oliveira, D. S. Leme, B. H. G. Barbosa, M. P. Rodarte, and R. G. F. A. Pereira, "A computer vision system for coffee beans classification based on computational intelligence techniques," *Journal of Food Engineering*, vol. 171, pp. 22–27, Feb 2016.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[5] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.

[6] J. Chai, H. Zeng, A. Li, and E. W. Ngai, "Deep learning in computer vision: A critical review of emerging techniques and application scenarios," *Machine Learning with Applications*, vol. 6, p. 100134, 2021.

[7] A. G. Costa, D. A. G. D. Sousa, J. L. Paes, J. P. B. Cunha, and M. V. M. D. Oliveira, "Classification of robusta coffee fruits at different maturation stages using colorimetric characteristics," *Engenharia Agrícola*, vol. 40, no. 4, pp. 518–525, Aug. 2020.

[8] S. Pradana-López, A. M. Pérez-Calabuig, J. C. Cancilla, M. Ángel Lozano, C. Rodrigo, M. L. Mena, and J. S. Torrecilla, "Deep transfer learning to verify quality and safety of ground coffee," *Food Control*, vol. 122, p. 107801, 2021.

[9] S.-J. Chang and C.-Y. Huang, "Deep Learning Model for the Inspection of Coffee Bean Defects," *Applied Sciences*, vol. 11, no. 17, 2021.

[10] P. Wang, H.-W. Tseng, T.-C. Chen, and C.-H. Hsia, "Deep Convolutional Neural Network for Coffee Bean Inspection." *Sensors & Materials*, vol. 33, 2021.

[11] M. A. Tamayo-Monsalve, E. Mercado-Ruiz, J. P. Villa-Pulgarin, M. A. Bravo-Ortíz, H. B. Arteaga-Arteaga, A. Mora-Rubio, J. A. Alzate-Grisales, D. Arias-Garzon, V. Romero-Cano, S. Orozco-Arias, G. Gustavo-Osorio, and R. Tabares-Soto, "Coffee Maturity Classification Using Convolutional Neural Networks and Transfer Learning," *IEEE Access*, vol. 10, pp. 42 971–42 982, 2022.

[12] A. Febriana, K. Muchtar, R. Dawood, and C.-Y. Lin, "USK-COFFEE Dataset: A Multi-Class Green Arabica Coffee Bean Dataset for Deep Learning," in *2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom)*, 2022, pp. 469–473.

[13] E. L. da Rocha, L. Rodrigues, and J. F. Mari, "Maize leaf disease classification using convolutional neural networks and hyperparameter optimization," in *Anais do XVI Workshop de Visão Computacional*. Porto Alegre, RS, Brasil: SBC, 2020, pp. 104–110. [Online]. Available: https://sol.sbc.org.br/index.php/wvc/article/view/13489

[14] B. Espejo-Garcia, I. Malounas, N. Mylonas, A. Kasimati, and S. Fountas, "Using EfficientNet and transfer learning for image-based diagnosis of nutrient deficiencies," *Computers and Electronics in Agriculture*, vol. 196, p. 106868, 2022.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[17] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[18] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation," *CoRR*, vol. abs/1801.04381, 2018. [Online]. Available: http://arxiv.org/abs/1801.04381

[19] A. Howard, M. Sandler, G. Chu, L. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for mobilenetv3," *CoRR*, vol. abs/1905.02244, 2019. [Online]. Available: http://arxiv.org/abs/1905.02244

[20] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

[21] J. Deng, W. Dong, R. Socher, L. L, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Miami, FL, USA: IEEE, June 2009, pp. 248–255.