Implementation of Image Processing and CNN for Roasted-Coffee Level Classification

Irfan Asfy Fakhry Anto[1], Jony Winaryo Wibowo[2], Taufik Ibnu Salim[3], Aris Munandar[4]
[1] Research Center for Data and Information Science, National Research and Innovation, Indonesia
[2,3,4] Research Center for Smart Mechatronic, National Research and Innovation, Indonesia

Article Info

Article history:

Received Apr 5, 2024
Revised Aug 18, 2024
Accepted Dec 14, 2024

Keyword:

Roast coffee level
roasted coffee
convolutional neural network
image processing
lab color model

ABSTRACT

The roasting process of coffee beans plays a crucial role in the development of chemicals responsible for the rich color and complex flavors characteristic of well-roasted coffee. One approach to understanding this process involves assessing the roast level, which varies in color from light to dark, with intermediate levels in between. In this study, image processing was performed using Convolutional Neural Networks (CNNs), a widely used method for image classification. The objective was to utilize the LAB color model and the CNN framework to classify the roast levels of coffee beans based on images from files or video streams. The study also details the hardware and software tools employed. A user-friendly graphical interface was developed to ensure ease of use, requiring minimal training for efficient operation. The research successfully designed, developed, and implemented an application for classifying coffee bean roast levels using two methods: LAB color model image processing and the CNN model. Consequently, the system can recognize roast levels based on the outputs from both the LAB model and the CNN model. This research represents a preliminary effort and requires further development to support more extensive studies. Ultimately, it serves as a foundation for future exploration and the application of embedded system-based solutions for assessing coffee bean maturity levels in alignment with Agtron classification standards.

Corresponding Author:

Irfan Asfy Fakhry Anto,
Research Center for Data and Information Science,
National Research and Innovation,
KST Samaun Samadikun, Bandung, 40135, Indonesia.
Email: irfa013@brin.go.id

1. INTRODUCTION

Coffee bean roasting is a pivotal step in the journey from raw coffee beans to the aromatic, flavorful brew millions worldwide enjoy. Roasting not only enhances the beans' flavor but also brings out their distinct aromas, acidity levels, and body, crucial elements in determining the quality and character of the final cup of coffee [1]. Additionally, the roasting process plays a significant role in the formation of compounds, contributing to the rich colour and complex flavours characteristic of well-roasted coffee [2]. Coffee bean roasting is a nuanced craft that requires skill, precision, and a deep understanding of the beans' characteristics. Roasters carefully monitor factors such as temperature, airflow, and roast time to achieve desired flavor profiles and ensure consistency in their product [3].

Furthermore, the Specialty Coffee Association (SCA) plays a pivotal role in shaping the global specialty coffee landscape, serving as a leading authority on coffee excellence, sustainability, and industry standards [4]. This association developed the Agtron Number International Standard, this numerical system assigns values to the color of roasted coffee beans, ranging from light to dark, with lower numbers indicating lighter roasts and higher numbers signifying darker roasts [5]. The Agtron can be used to classify based on color, which differentiates the flavor of the coffee.

Classifying coffee bean roasting methods is essential for understanding the diverse array of flavors and characteristics present in coffee. Various methodologies have been developed to categorize roasting techniques, each offering unique insights into the process and its outcomes. One approach involves considering the degree of roast, which encompasses a spectrum ranging from light to dark roasts, with intermediate levels in between [6]. Another classification method focuses on the roasting equipment used, distinguishing between traditional drum roasters, fluidized bed roasters, and newer technologies such as infrared and microwave roasters [7]. Moreover, some classification systems incorporate sensory evaluations to assess roast profiles based on attributes like aroma, acidity, body, and flavor intensity [8].

Advancements in roasting technology and the emergence of specialty coffee culture have sparked innovation in roasting techniques, leading to a diverse range of roast styles tailored to cater to varying preferences among coffee enthusiasts [9]. For example, on the market, there are available products for color classification of roasted coffee beans such as Color Track RT [10]. This utilizes advanced spectrophotometric or laser sensor analysis to assess the colour development of coffee beans during roasting with unparalleled accuracy and precision. This tool uses the LAB color model as colour assessment.

Several approaches have been proposed to solve roasting coffee levels. A comprehensive review of related works about research trends, methods, technology, and evaluation metrics in coffee bean roasting level classification, is provided in our previous work [11]. However, motivated by previous work, image processing was performed following Convolutional Neural Networks (CNN) which was the most method used for image classification. Accordingly, this research subject is to implement the LAB color model and CNN models, to classify the color or Agtron level of coffee bean roasting images in a file or video stream. These models are applied with A graphical user interface (GUI) and the Raspberry Pi 4 within the camera as an embedded system.

In this paper, it presents a prototype system for real-time roast-level classification as preliminary research. The reason for choosing an embedded system is can be more easily integrated into a roaster machine in future rather than using a smartphone. This paper sections as follows. Section 2 presents a summary of related studies. Section 3 describes the materials and method, including the hardware and software components used. Section 4 presents the results and discussion. The last section concludes the article and directions for future research in this area.

2. RELATED STUDIES

2.1. Roasted Coffee Level Classification

Table 1. Agtron Roasting profiles

Agtron Code	Color	Degree	Agtron Code	Color	Degree
#R95		Very light	#R55		Medium
#R85		Light	#R45		Moderately dark
#R75		Moderately light	#R35		Dark
#R65		Medium Light	#R25		Very dark

Roasting time and temperature determine the degree of roasting that is required to achieve the essential chemical reactions without burning the beans and affecting the beverage's flavor [12]. The Specialty Coffee Association of America (SCAA) issued an Agtron Roast Color Classification System [13] used to monitor the degree of roasting. The Agtron number divides it into intervals of values. Each of these numbers represents a temperature range; the lower the number, the more roasting there is. Agtron constructed discs of various hues based on the standards outlined in the scale and built an infrared spectrometer to measure the degree of roasting.

As a result, SCAA standardized the roasting control process by employing these disks and the infrared [14]. Agtron roasting profiles in coffee varied from very light to very dark as shown in Table 1 [15]. Therefore, in this study, the use of the SCA guide, particularly the Coffee Roast Color Guide RG-770, is deemed feasible when applying the Agtron roasting profile and utilizing coffee bean samples in the form of single beans [16], [17] within a roasting process determined by roasting time and temperature [18], [19].

2.2. LAB Color Model

T The LAB color model is a color space that represents colors independently of devices (such as cameras or displays) and is based on human vision. It consists of three components: L* (lightness), represented on a vertical axis with values ranging from 0 (black) to 100 (white); a* (green to red), where positive a* values indicate red and negative a* values indicate green; and b* (blue to yellow), where positive b* values indicate yellow and negative b* values indicate blue [20].In image processing with the LAB colour model, adjusting the L* channel aids overall brightness correction, while modifying the a* and b* channels addresses colour imbalances. LAB's perceptual uniformity is leveraged for image segmentation through thresholding in the L* channel or K-Means clustering in LAB space. Colour-based object recognition benefits from LAB histograms, providing colour information unaffected by lighting variations. Enhancing features involves independent processing of the L* channel for contrast improvement, and selective colour enhancement is achieved by manipulating the * and b* channels. This paper uses the LAB model according to the Agtron scale conversion to the LAB colour model [21], as shown in Figure 1.

Figure 1. Agtron scale conversion to LAB color model [21]

2.3. CNN Model

CNN is a type of artificial neural network architecture that is commonly used for image processing and visual pattern recognition tasks [22]. CNNs are effective in addressing image recognition problems since can extract hierarchical features from image data. The illustration of CNN architecture is shown in Figure 2. The main components of a CNN [23]:

(a) Convolutional Layers: Convolutional layers are responsible for extracting features from the input image by using convolution filters or kernels. These filters move across the image to detect patterns such as edges, corners, or textures.

(b) Pooling Layers: Pooling layers (generally using max pooling) are used to reduce the spatial dimension of the feature representation and retain the most important information. This helps reduce the number of parameters that need to be learned by the model and improves computational sustainability.

(c) Fully Connected Layers: After multiple convolution and pooling layers, the results are then processed by the fully connected layer (sometimes called the Dense layer). This layer is responsible for combining the extracted features for final decision-making. Activation Functions: Activation Functions such as ReLU (Rectified Linear Unit) are often used to add an element of non-linearity to the model, which helps the model learn more complex patterns.

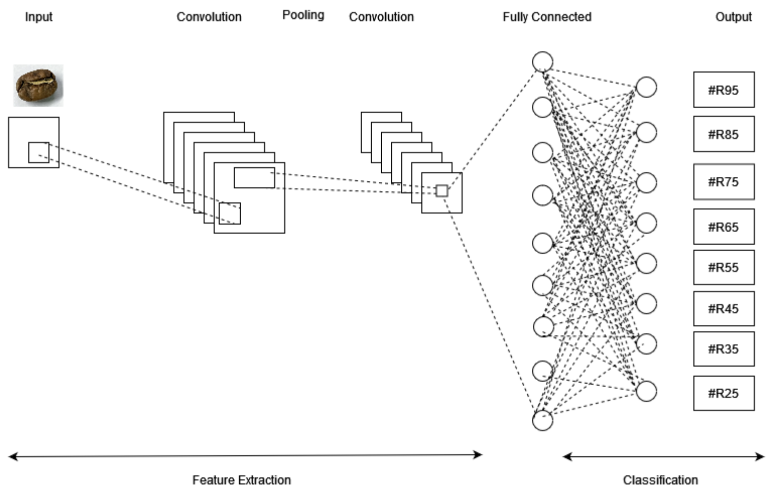


Figure 2. Ilustration CNN Model for Roasted Coffee Level Classification

2.3.1. HyperResNet Model

HyperResNet is a convolutional neural network architecture proposed by [24], specifically designed for high-resolution image classification tasks. This architecture in Figure 3 builds upon the traditional ResNet model by incorporating hypercolumns, which are feature representations obtained by combining feature maps from different layers of the network. HyperResNet aims to leverage the hierarchical features extracted at multiple scales to improve the accuracy of image classification, particularly for high-resolution images where fine-grained details are crucial. This research has chosen HyperResNet since designed for high-resolution image classification.

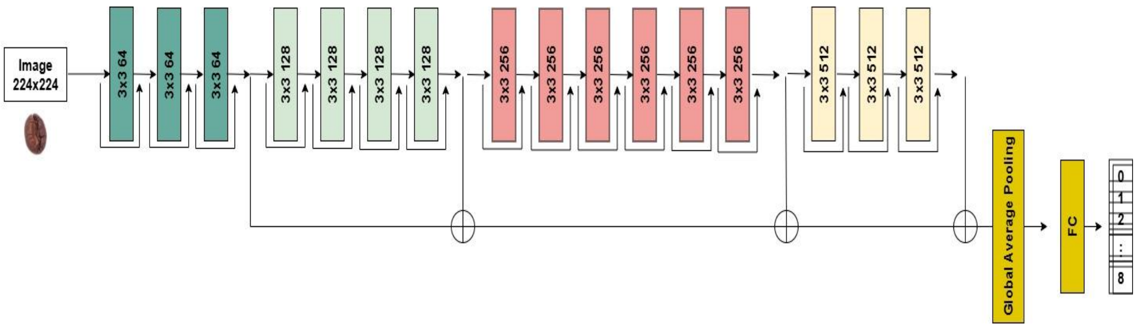


Figure 3. Ilustration of HyperResNet Architecture [24]

2.4. GAP Analysis

A brief overview of recent other approach for GAP analysis can be seen in Table 2. Based on Table 2, this study differs from another recent research. Our study employs two methods: LAB model and CNN (HyperResNet) for classification. Additionally, our study utilizes its own dataset, featuring an eight-level classification system based on the Agtron number SCA standard. We also developed a prototype system comprising an embedded system using Raspberry Pi. The real-time embedded system is a significant strength, offering both flexibility and scalability. In contrast, other studies three-to-four-level classification systems, typically without real-time applications or integration into embedded systems. Furthermore, some papers primarily focus on model evaluation, which may restrict their broader applicability, since our study provide prototyope system capable to upload the model into the prototype system.

Table 2. GAP Analysis with other approach

Reference	Key Findings	Gaps Identified
[25]	Achieved a classification accuracy of 97.5% using Backpropagation Neural Network for single coffee beans.	• Color classification is using four levels of roasting instead of using Agtron - SCA Standards. • Web based application for GUI. • Method not using LAB color identification
[26]	The system achieved an accuracy of 94.79% using MobileNetV2 and an accuracy of 90.40% using VGG19 in classifying the different types.	• Color classification is using four levels of roasting instead of using Agtron - SCA Standards. • Only experiment two model, none prototype or application for GUI. • Method not using LAB color identification
[27]	Achieved a classification accuracy of 97.5% using convolutional neural network for single coffee beans.	• Color classification is using four levels of roasting instead of using Agtron - SCA Standards. • Only experiment one model, none application for GUI. • Method not using LAB color identification
[28]	Achieved a classification accuracy of 90.30% using least squares support vector machine (LS-SVM) in classifying the different types.	• Color classification is using seven roasting degrees of Roast Color Classification System (Agtron/SCAA) • A laboratory HSI system is used for experiment, it need to connect with a computer with the Spectral-Cube data acquisition software (Isuzu Optics) • Method not using LAB color identification

3. RESEARCH MATERIALS AND METHODS

In this research, an overview of research materials and methods is shown in Figure 4. First, collected my dataset by capturing images of roasted coffee beans as described in Section 3.1. The dataset was divided into 80% for training data, 10% for validation, and 10% for testing.

The next step is modelling image preprocessing and CNN. Moreover, GUI design and development on personal computers. This step is parallel work with embedded device assembly. The embedded device uses the Raspberry Pi 4 and is equipped with a camera module to capture real-time video streams for coffee beans.

For the CNN model is needed training process. The training was conducted on cloud computing using *MAHAMERU BRIN HPC* which resulted in the output model. The output model file was imported into a personal computer for testing and integrated with Graphical User Interface (GUI). Once the training and testing phases conclude, all of the modules and scripts are able onto the embedded device. In the end, after all, is ready and then able to import into the embedded system.

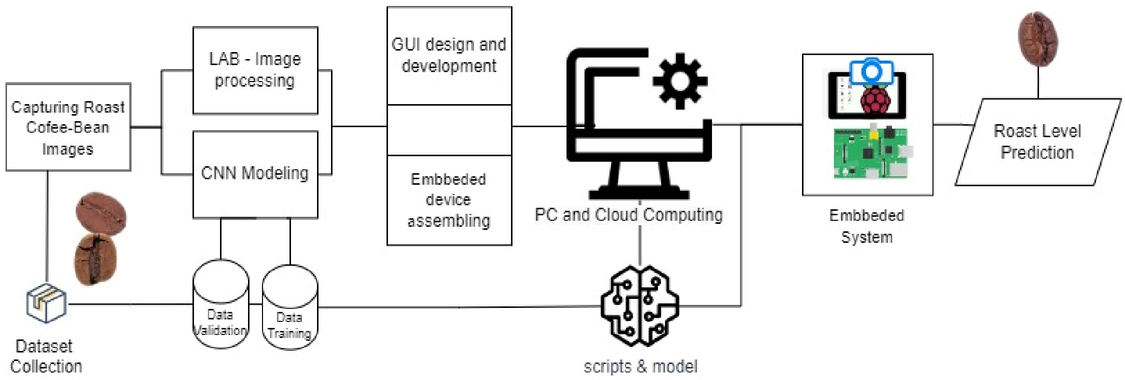
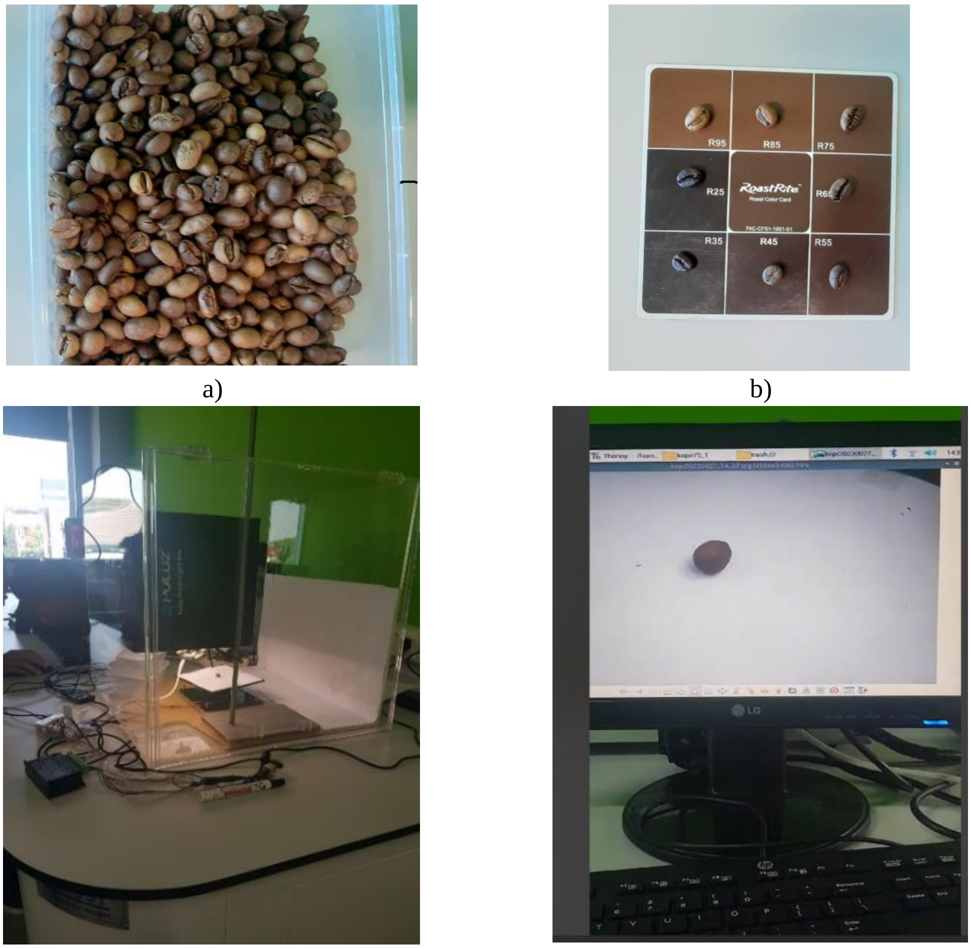


Figure 4. An Overview of Research Materials and Methods

3.1. Capturing Roast Coffee-Bean Data



a) b)

c) d)

Figure 5. Dataset collection equipment and tool setup a) Coffee beans b) Agtron 770 standard c) Mini studio light box with Brio 4K Webcam camera inside d) PC and sample image

This research gathered its data. For experimental setup for capturing the dataset is shown in Figure 5. Coffee beans that have undergone the roasting process were collected before being photographed as shown in Figure 5 (a). The experiment tool comprises Agtron 770 standard as shown in Figure 5 (b), a personal computer in Figure 5 (d) connected to a Brio 4K Webcam camera inside the photo light box. To ensure optimal conditions, a Puluz photo light box is also necessary as shown in Figure 5 (c).

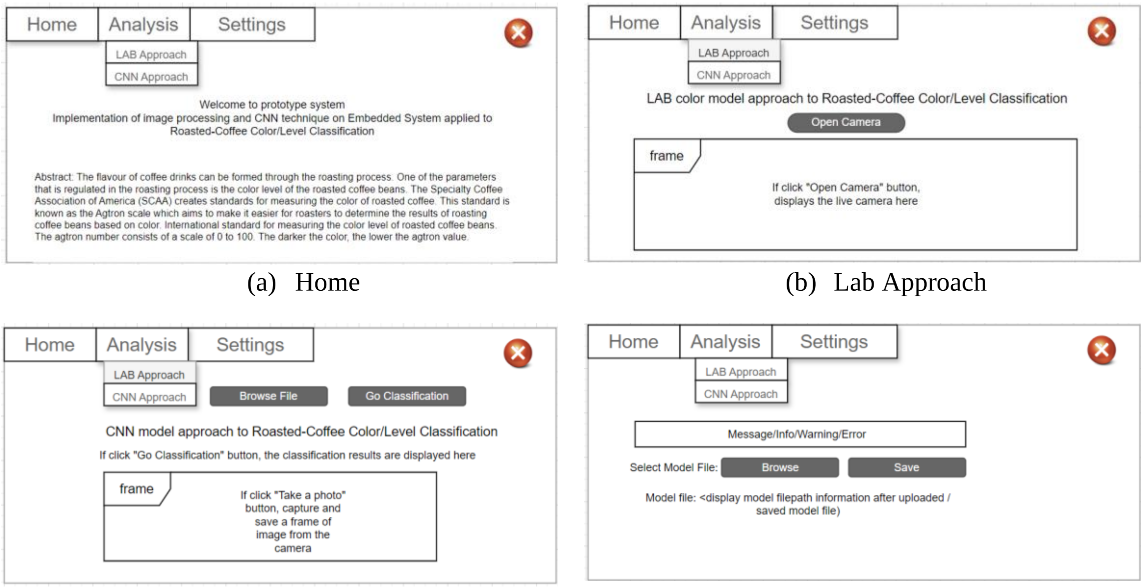
The initial stage involves comparing each coffee bean with the Agtron 770 standard as shown in Figure 5 (b). The photography procedure entails using a camera sensor to capture images, which are subsequently categorized by class and stored in the image file locally. This process is repeated for each coffee bean. The dataset has been categorized into eight classes, as detailed in Section 4.1. However, in this research, the results of this dataset collection will solely be utilized to implement the CNN model only, as the LAB model does not necessitate a training process.

3.2. GUI Design

To make it convenient for users to operate the system, a graphical user interface (GUI) as shown in Figure 6, is designed to provide an easily accessible interface for human-machine interaction. The GUI consists of 4 other menus Home, Analysis, Settings and Exit.

In consequence, the Analysis menu is divided into 2 submenus, including "Lab Approach" and "CNN model Approach".When the "Home" menu is clicked, it will display descriptive information about this system, as illustrated in Figure 6 (a). If the "LAB Approach" is selected, a label button and an "Open Camera" option will be displayed as illustrated in Figure 6 (b). Clicking this button will activate the live camera feed.

In the recognition process using the LAB method, testing data is directed towards the camera, and the application captures images in real time. The data is captured via the camera and there is a function that conducts the recognition process. It then creates a barrier around the coffee bean object, followed by calculating the values of L, A, and B as described in section 4.2. For the CNN method as illustrated in Figure 6 (c), the selected model must undergo prior training. This application allows uploading the output-trained models in menu Settings as illustrated in Figure 6 (d). Testing data in the CNN method relies on browsing a file and then clicking the “Go Classification” button. The predicted label will be displayed on the frame below the button.



(a) Home

(b) Lab Approach

(c) CNN Approach

(d) Settings

Figure 6. Mockup of GUI

4. RESULTS AND DISCUSSION

4.1. Dataset Collection

The dataset comprises images of roasted coffee beans, each labelled individually through the process of photographing and labelling, resulting in a comprehensive collection of datasets. A summary of the dataset collection results is presented in Table 3. Before its utilization in this research, the dataset underwent preprocessing to enhance its quality and diversity.

Table 3. Dataset Collection Summary

Class	Number files		
	Training Files	Validation File	Test file
R25	368	46	46
R35	531	66	66
R45	470	59	59
R55	516	64	64
R65	468	58	58
R75	524	65	65
R85	546	68	68
R95	668	83	83
total	4091	509	509

Data augmentation techniques were employed to enrich the variety and quantity of images. Operations such as cropping, flipping, rotating, and aligning were applied to the images. This augmentation process contributes to improved image representation. A selection of sample images from the dataset is depicted in Figure 7.

This dataset encompasses a total of 5109 image files as shown in Figure 8 and [29], showcasing coffee beans with various colours, categorized into 8 Agtron color classifications, as detailed in Table 3. It is segmented into three subsets: training, validation, and testing sets. Approximately 80% of the data, equivalent to 4091 images, is allocated for training purposes. The validation set comprises 10% of the data, totaling 509 images, while the remaining 10% of the data equivalent to 509 images is reserved for testing.

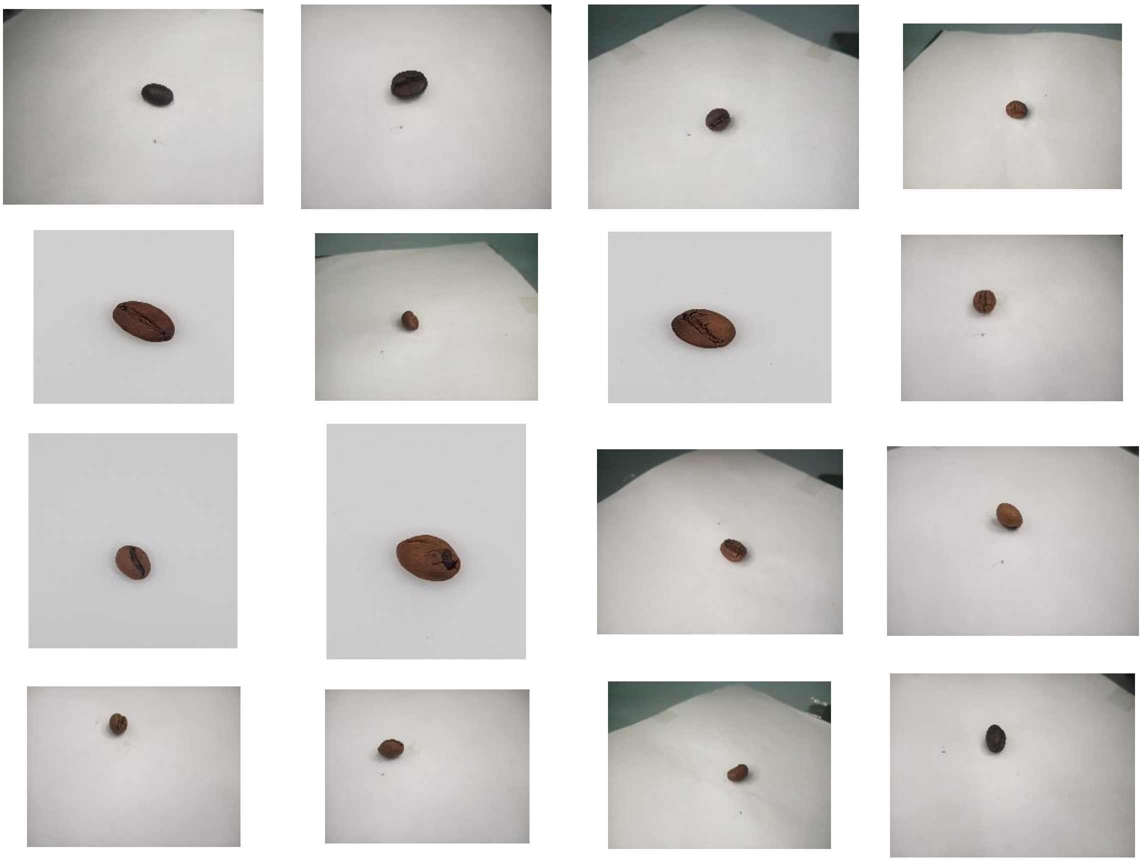


Figure 7. Sample original images from dataset

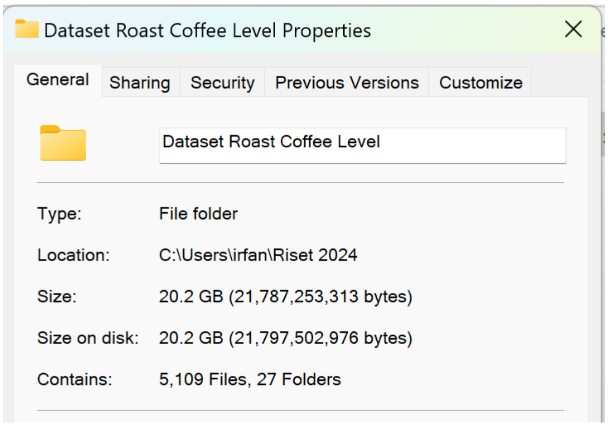


Figure 8. Dataset Properties

4.2. Implementation of LAB Model

The implementation of the LAB model for this research used the main library, OpenCV, along with support libraries such as Numpy, Pillow, and Colormath. Every frame from the video stream is analyzed. A frame is captured in LAB colour, and then an image segmentation process is carried out, dividing the image into segments or objects based on the homogeneity of pixel characteristics.

The first step is to read the image and convert it from LAB to HSV color space. This conversion task uses the `cv2.cvtColor()` function, with the `cv2.COLOR_LAB2HSV` flag employed to convert from LAB to HSV colour space. Next, a list of colours is defined based on Figure 1. In this case, it's an array list containing the relationship between LAB and Agtron values, as shown in the script snippet in Figure 9.

The next step involves the segmentation process, where the lower and upper bounds of the L color are defined. The lower bound is set as (L-5, A, B), and the upper bound as (L+5, A, B). For this experiment, a difference of 5 is used for each class for the L value. This number may vary depending on the specific case or experiment.

The subsequent task is to create a mask based on this range as illustrated in Figure 10. Based on the results of this masking, a detected area box is obtained according to the LAB colour defined. All of these steps are performed in real time and repeated for each frame in the video.

```
agtron_to_lab = {
    "80" : [57, 8, 16],
    "70-60" : [42, 5, 15],
    "50-55" : [37, 5, 13],
    "45-50" : [31, 4, 9],
    "40-45" : [29, 3, 8],
    "40-35" : [32, 3, 8],
    "35-30" : [20, 0, 4],
    "30-25" : [18, 0, 4],
    "25-15" : [6, -2, 3]
}
```
Figure 9. Sample script for defined an array

```
mask = cv2.inRange(frame, lowerLimit, upperLimit)
mask_ = PIL.Image.fromarray(mask)
```
Figure 10. A piece of script for making a mask

4.3. Implementation of HyperRestNet Model

Table 4. Hyperparameters properties

Hyperparameters	Value
batch_size	32
learning-rate	0.001
epochs	10
optimizer	SGD
SGD momentum	0.9
SGD weigh-decay	0.005

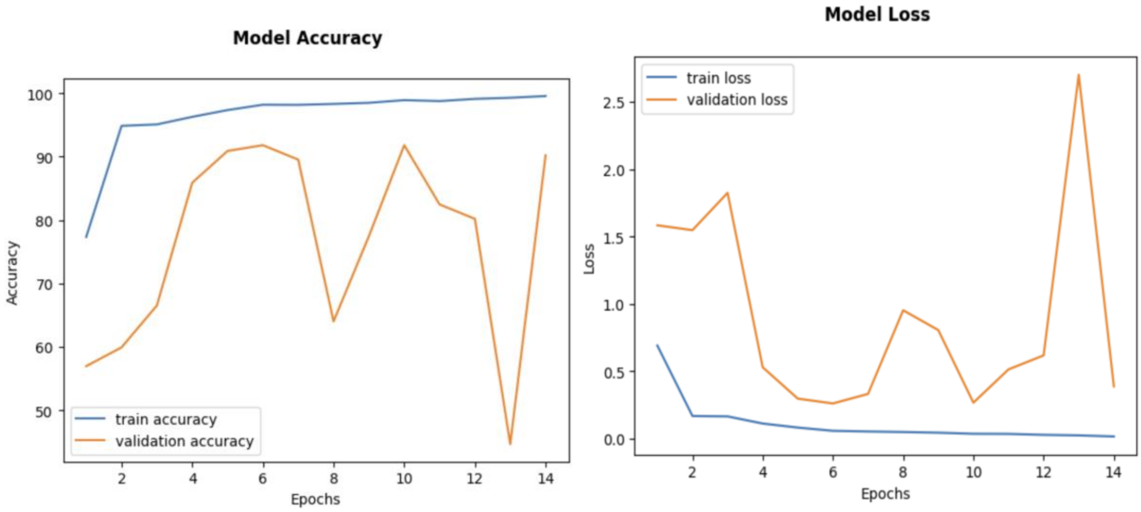


Figure 11. Progression of (a) Accuracy, (b) Loss for HyperRestNet Model

The HyperRestNet modeling of roast level coffee-bean was performed at *HPC Mahameru*, provided access enabling efficient training and inference processes for generating model. The model was trained on the dataset. Training process using the PyTorch [30] library and HyperResNet class module. The hyperparameters were used during this process as shown in Table 4. Figure 11 shows the progress of accuracy and loss of training data. The results of research conducted for 14 epochs show that the best *train_accuracy* results were obtained at the 14th epoch as is as 99.552%, while the best *validation_accuracy* was at the 10th epoch as is as 91.8%. For the best loss result or close to 1, the train error value (*train_loss*) is 0.015 at the 14th epoch, while for validation loss value (val_loss) is 0.26 at the 10th epoch.

The final training step is executed after completing the training process, saving the model output into a file. Save it to a file in *pth* format as name *HyperResNet.pth*. This process utilizes the torch. Save function, which employs Python's pickle utility for serialization. Models, tensors, and dictionaries of all types of objects can be stored using this function. Finally, this output file will be exported or copied into the GUI program to menu Settings in GUI program.

Once the training step is complete, the model output file can be used for testing purposes. This research did not include additional comparisons or experiments to determine the best model but instead focused on integrating the output model file into the system. The system includes an option in the Settings menu to replace the output model file. Consequently, further training and testing result in a model with better accuracy, the new output file can be uploaded to the system's local directory.

4.4. Implementation of GUI and Embbeded System

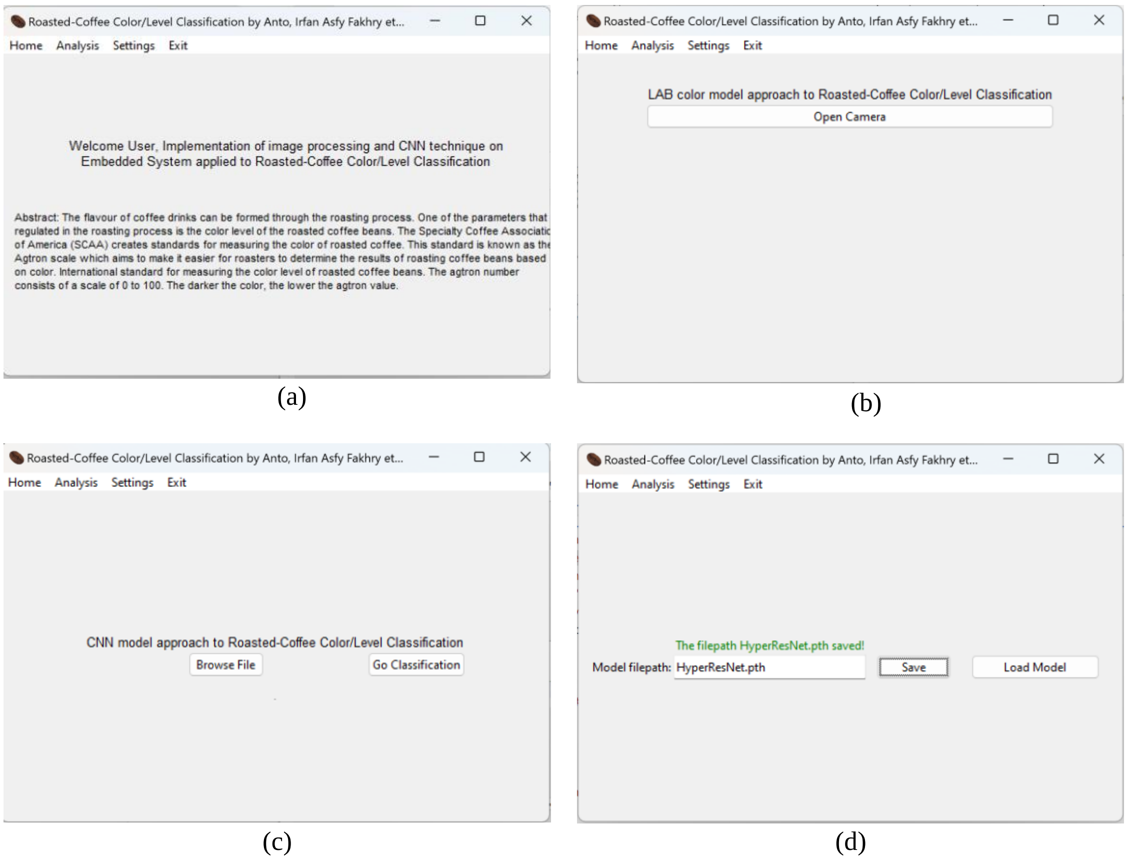


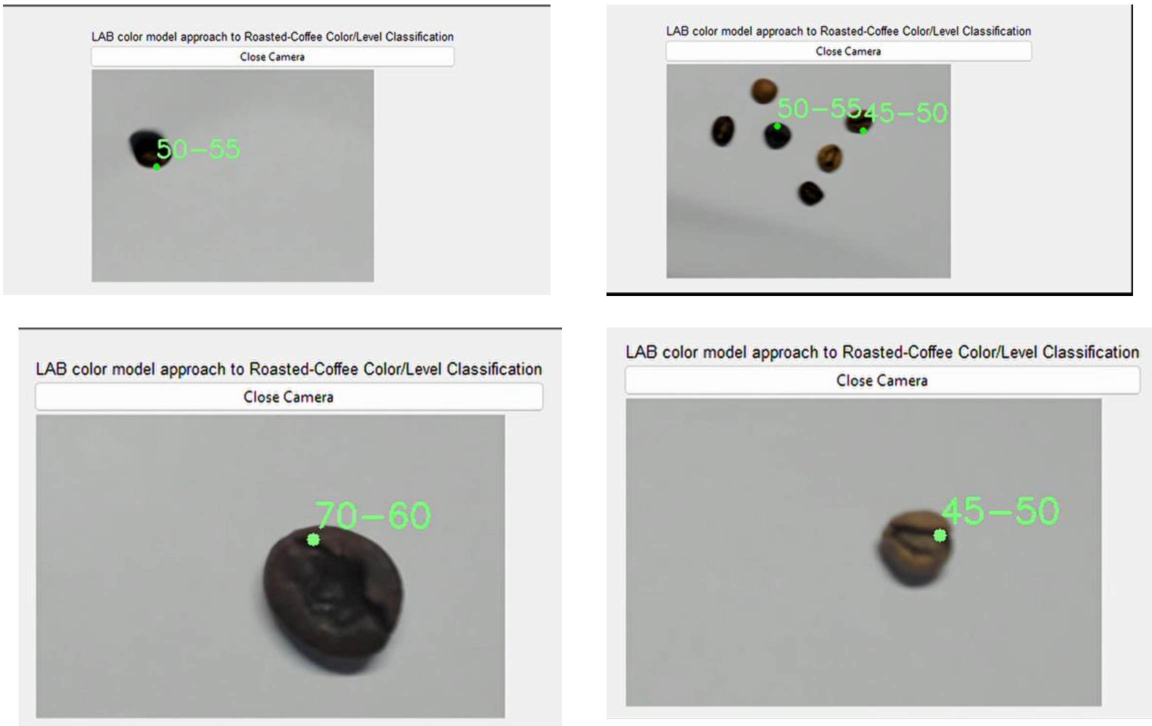
Figure 12. GUI Implementation

For application design and development tools use using Personal Computer on Windows 11 with i9-11900H @ 2.50GHz (16 CPUs), ~2.5GHz, 16 GB RAM and an NVIDIA GeForce RTX 3050 Ti. The result of the GUI is shown in Figure 13 and the complete configuration of the embedded device with other components is shown in Figure 14. The GUI was developed using the Windows Forms application within the Python programming language and used Tkinker as the main library module. Employing Sublime Text as the

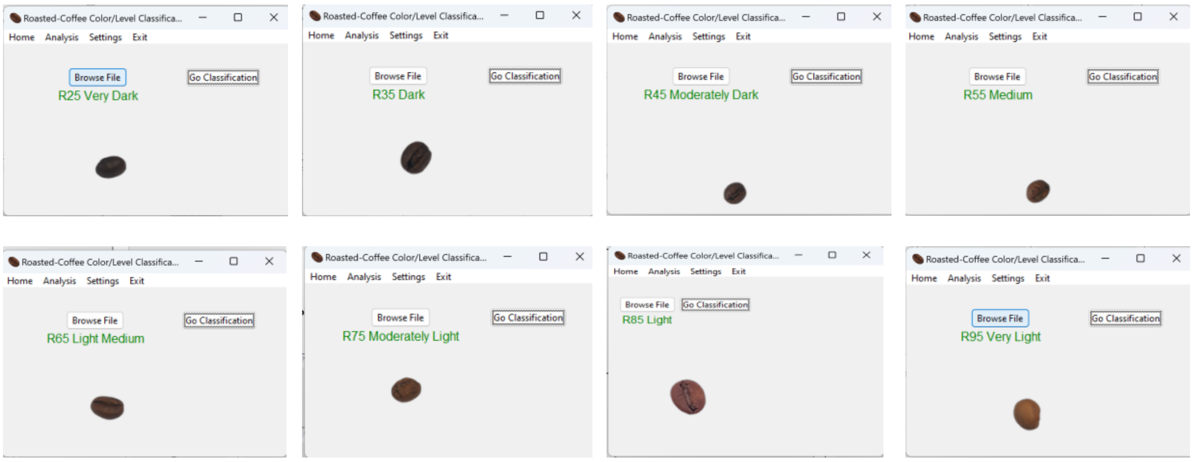
Integrated Development Environment (IDE) facilitated seamless integration and streamlined development for this research. Utilizing Windows Forms technology, the GUI offers an intuitive and adaptable platform for creating a sophisticated user interface tailored for users.

Figure 13. Complete layout of the connections of Raspberry, Monitor, Camera, Power Supply and other component.

By using this GUI was used for testing and experiments in this research. The GUI facilitates the analysis of two approaches: (a) the LAB model and (b) the CNN approach. LAB uses the live camera for the classification process. For CNN Approach enables saving the output of the model file after uploading it to the local directory using the settings menu. A comparison of the different experiments from the two models when classifying roast level in the video stream and uploading the image is shown in Figure 15. Based on this experiment, the system can show some coffee beans are selected and able to predict the roast level or Agtron number. Therefore, all functionality of the system is working well, and the implementation of the system can recognize the roast level of coffee beans according to the LAB model and CNN model output file given.



(a) Experiments using LAB model



(b) Experiments using CNN model

Figure 14. Comparison of two different methods, a) LAB Approach and b) CNN Approach.

4.5. Discussion

This research has been designed and developing for implemented the system for roast bean level classification. Capturing the image roast level for creating own dataset is using for training HyperResNet model only because LAB model it's no need training process. Therefore, design and developed GUI with embedded device assembling has been done successfully as embedded system.

This embedded system can perform coffee bean level recognition using two methods, namely the LAB approach and the CNN model approach. Comparing this result with existing products like Color Track RT as the following:

- Color Track RT has a laser sensor and LAB model, this research uses a camera sensor instead of a laser sensor, and analysis using not only the LAB colour model but also CNN techniques as alternative solutions.
- At cost, the camera sensor is cheaper than the laser sensor. Also, using CNN can better follow developments in object or colour recognition or coffee-bean technology trends in the future.

To compare our work with others, as shown in Table 6, the first difference lies in the dataset type, as our research utilizes a custom dataset. Additionally, we developed a prototype system using Raspberry Pi with a GUI for embedded real-time classification, making it suitable for industrial applications and offering an improved user experience through a custom interface. Our research primarily focuses on model implementation rather than conducting comprehensive model evaluations or extensive statistical analysis of the experimental results. However, the prototype system includes a Settings menu, allowing users to upload CNN model files and update the model with future experimental results. The findings of this study represent preliminary research and require further development. Addressing these limitations in future research could lead to the development of other CNN models.

Table 5 Comparison Proposed system with other work

Criteria	Proposed System	Other Work	Comparison Insights
Data Handling	Create new dataset, 5109 images; extensive data augmentation	Many studies use public datasets with limited classes	Superior due to tailored dataset and comprehensive preprocessing.
Real-Time Capability	Achieved using Raspberry Pi with GUI	Limited real-time implementations	Excels with embedded real-time classification, suitable for industry applications.
Cost Efficiency	Utilized a camera sensor (cheaper)	HSI system, PC, Laser sensors (high-cost)	Cost-effective while maintaining comparable accuracy.
User-Friendliness	GUI-based, intuitive interface	Often lacks direct user interface	Better user experience through custom GUI.

5. CONCLUSION

This research designed GUI, developed a prototype system, and implemented model for classifying coffee bean maturity levels using two methods: LAB model image processing and a CNN model. All stages of the process, from data collection and design to testing, were successfully carried out. The data collection process resulted in a dataset used for CNN modeling. During the design phase, attention was given not only to the GUI but also to the assembly of the embedded device. The training process exclusively utilized the dataset for the CNN model. The prototype system was subsequently tested as an embedded device.

Based on the test results, both the LAB model and the CNN model were able to predict image classes in alignment with Agtron color classes. All system functionalities were thoroughly tested and performed effectively. Consequently, the system is capable of recognizing the roast level of coffee beans based on the output files from the LAB and CNN models. Additionally, the application employs the CNN model to monitor technological trends in the coffee industry.

Despite these achievements, this research requires further development. For instance, future work should include more comprehensive statistical analysis of the experimental results. Subsequent studies may focus on enhancing the training and testing of CNN models to address the limitations identified in this research. The findings of this study provide a foundation for the utilization of coffee image Agtron recognition technology. We hope that this research inspires further exploration and application of embedded system-based solutions for measuring the maturity levels of coffee beans and generating flavor profiles corresponding to Agtron classes.

ACKNOWLEDGMENTS

Support for this research was provided by the Indonesian National Research and Innovation Agency, for which the authors are especially grateful. The computation in this work has been done using the facilities of MAHAMERU BRIN HPC, National Research and Innovation Agency of Indonesia (BRIN).

REFERENCES

[1] B. Folmer, *The Craft and Science of Coffee*. London: Academic Press - Elsevier, 2017.

[2] A. N. Gloess *et al.*, "Comparison of nine common coffee extraction methods: instrumental and sensory analysis," *European Food Research and Technology*, vol. 236, no. 4, pp. 607–627, Apr. 2013, doi: 10.1007/s00217-013-1917-x.

[3] E. K. Bekedam, M. J. Loots, H. A. Schols, M. A. J. S. Van Boekel, and G. Smit, "Roasting Effects on Formation Mechanisms of Coffee Brew Melanoidins," *J Agric Food Chem*, vol. 56, no. 16, pp. 7138–7145, Aug. 2008, doi: 10.1021/jf800999a.

[4] The Specialty Coffee Association (SCA), "About." Accessed: Nov. 21, 2023. [Online]. Available: https://sca.coffee/about

[5] S. Schenker, C. Heinemann, M. Huber, R. Pompizzi, R. Perren, and R. Escher, "Impact of Roasting Conditions on the Formation of Aroma Compounds in Coffee Beans," *J Food Sci*, vol. 67, no. 1, pp. 60–66, Jan. 2002, doi: 10.1111/j.1365-2621.2002.tb11359.x.

[6] M. Bozzola *et al.*, *The Coffee Guide Fourth edition TRADE IMPACT FOR GOOD*, 4th ed. Geneva: International Trade Centre, 2021. Accessed: Feb. 11, 2024. [Online]. Available: https://www.researchgate.net/publication/355928597_The_Coffee_Guide_Fourth_Edition

[7] J. Avelino *et al.*, "The coffee rust crises in Colombia and Central America (2008–2013): impacts, plausible causes and proposed solutions," *Food Secur*, vol. 7, no. 2, pp. 303–321, Apr. 2015, doi: 10.1007/s12571-015-0446-9.

[8] I. Santoso, S. A. Mustaniroh, and A. Choirun, "Methods for quality coffee roasting degree evaluation: A literature review on risk perspective," in *IOP Conference Series: Earth and Environmental Science*, IOP Publishing Ltd, Dec. 2021. doi: 10.1088/1755-1315/924/1/012058.

[9] B. Odžaković, N. Džinić, Z. Kukrić, and S. Grujić, "Effect of roasting degree on the antioxidant activity of different Arabica coffee quality classes," *Acta Sci Pol Technol Aliment*, vol. 15, no. 4, pp. 409–417, Dec. 2016, doi: 10.17306/J.AFS.2016.4.39.

[10] Coffee Laboratory, "ColorTrack RT." Accessed: Nov. 14, 2023. [Online]. Available: https://www.coffeelabequipment.com/COLORTRACKRT.html

[11] I. A. F. Anto, A. Munandar, J. W. Wibowo, T. I. Salim, and O. Mahendra, "Coffee Bean Roasting Levels Detection: A Systematic Review," in *2023 IEEE 7th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, IEEE, Nov. 2023, pp. 146–151. doi: 10.1109/ICITISEE58992.2023.10404775.

[12] C. Somporn, A. Kamtuo, P. Theerakulpisut, and S. Siriamornpun, "Effects of roasting degree on radical scavenging activity, phenolics and volatile compounds of Arabica coffee beans (Coffea arabica L. cv. Catimor)," *Int J Food Sci Technol*, vol. 46, no. 11, pp. 2287–2296, Nov. 2011, doi: 10.1111/j.1365-2621.2011.02748.x.

[13] The Specialty Coffee Association (SCA), "About Us." Accessed: Mar. 07, 2024. [Online]. Available: https://sca.coffee/about

[14] C. Q. da Silva *et al.*, "Risk assessment of coffees of different qualities and degrees of roasting," *Food Research International*, vol. 141, p. 110089, Mar. 2021, doi: 10.1016/j.foodres.2020.110089.

[15] F. de C. Pires, R. G. F. A. Pereira, M. R. Baqueta, P. Valderrama, and R. Alves da Rocha, "Near-infrared spectroscopy and multivariate calibration as an alternative to the Agtron to predict roasting degrees in coffee beans and ground coffees," *Food Chem*, vol. 365, p. 130471, Dec. 2021, doi: 10.1016/j.foodchem.2021.130471.

[16] N. Caporaso, M. B. Whitworth, and I. D. Fisk, "Prediction of coffee aroma from single roasted coffee beans by hyperspectral imaging," *Food Chem*, vol. 371, p. 131159, Mar. 2022, doi: 10.1016/j.foodchem.2021.131159.

[17] N. Caporaso, M. B. Whitworth, C. Cui, and I. D. Fisk, "Variability of single bean coffee volatile compounds of Arabica and robusta roasted coffees analysed by SPME-GC-MS," *Food Research International*, vol. 108, pp. 628–640, Jun. 2018, doi: 10.1016/j.foodres.2018.03.077.

[18] Mario Fernández-Alduenda *et al.*, "A System to Assess Coffee Value Understanding the Specialty Coffee Association's Coffee Value Assessment," 2024. [Online]. Available: https://sca.coffee/research/coffee-standards

[19] V. V. Freitas *et al.*, "Impact of different roasting conditions on the chemical composition, antioxidant activities, and color of Coffea canephora and Coffea arabica L. samples," *Heliyon*, vol. 9, no. 9, p. e19580, Sep. 2023, doi: 10.1016/j.heliyon.2023.e19580.

[20] B. C. K. Ly, E. B. Dyer, J. L. Feig, A. L. Chien, and S. Del Bino, "Research Techniques Made Simple: Cutaneous Colorimetry: A Reliable Technique for Objective Skin Color Measurement," *Journal of Investigative Dermatology*, vol. 140, no. 1, pp. 3-12.e1, Jan. 2020, doi: 10.1016/j.jid.2019.11.003.

[21] Coffee Laboratory, "ColorTrack Coffee Color Selector." Accessed: Oct. 14, 2023. [Online]. Available: https://www.coffeelabequipment.com/Color_Selector_cl.pdf

[22] G. Xie *et al.*, "Generating Neural Networks for Diverse Networking Classification Tasks via Hardware-Aware Neural Architecture Search," *IEEE Transactions on Computers*, vol. 73, no. 2, pp. 481–494, Feb. 2024, doi: 10.1109/TC.2023.3333253.

[23] J. I. Pilataxi, J. E. Zambrano, C. A. Perez, and K. W. Bowyer, "Improved Search in Neuroevolution Using a Neural Architecture Classifier With the CNN Architecture Encoding as Feature Vector," *IEEE Access*, vol. 12, pp. 11987–12000, 2024, doi: 10.1109/ACCESS.2024.3355804.

[24] Mamadou Keita, "HyperResNet," Jul. 2022. Accessed: Nov. 12, 2023. [Online]. Available: https://github.com/Mamadou-Keita/HyperResNet

[25] T. H. Nasution and U. Andayani, "Recognition of Roasted Coffee Bean Levels using Image Processing and Neural Network," *IOP Conf Ser Mater Sci Eng*, vol. 180, p. 012059, Mar. 2017, doi: 10.1088/1757-899X/180/1/012059.

[26] H. S. Metha, K. Kusrini, and D. Ariatmanto, "Classification of types Roasted Coffee Beans using Convolutional Neural Network Method," *Sinkron*, vol. 8, no. 2, pp. 846–851, Mar. 2024, doi: 10.33395/sinkron.v8i2.13517.

[27] N. K. Naik and P. K. Sethy, "Roasted Coffee beans Classification based on Convolutional Neural Network," in *2022 International Conference on Futuristic Technologies (INCOFT)*, IEEE, Nov. 2022, pp. 1–3. doi: 10.1109/INCOFT55651.2022.10094378.

[28] B. Chu, K. Yu, Y. Zhao, and Y. He, "Development of Noninvasive Classification Methods for Different Roasting Degrees of Coffee Beans Using Hyperspectral Imaging," *Sensors*, vol. 18, no. 4, p. 1259, Apr. 2018, doi: 10.3390/s18041259.

[29] I. A. F. Anto, J. W. Wibowo, A. Munandar, and T. I. Salim, "Dataset citra roast-level Biji Kopi berdasarkan skala Agtron." Accessed: Dec. 16, 2023. [Online]. Available: https://hdl.handle.net/20.500.12690/RIN/CYFYAQ

[30] J. Ansel *et al.*, "PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation," in *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, New York, NY, USA: ACM, Apr. 2024, pp. 929–947. doi: 10.1145/3620665.3640366.