Journal of Applied Science, Engineering, Technology, and Education *ISSN: 2685-0591 (online)*
Vol. 4 No. 1 (2022)
https://doi.org/10.35877/454RI.asci842

Coffee Granularity Classification using Convolutional Neural Network (CNN)

Franky Leonard* & Habibullah Akbar

Faculty of Computer Science, Esa Unggul University, Indonesia

Abstract

Coffee granularity is important for micro, small, and medium industries, in the determination of various beverage products. Traditionally, the identification of its graininess is carried out through the human visual system. Therefore, this study aims to classify coffee granularity levels using Convolutional Neural Networks (CNN). The proposed CNN architecture, i.e., AlexNet model have seven layers, namely three convolutional, two max-pooling, and two hidden segments, respectively. A dataset of 1039 grinder images were obtained from several coffee drink stores in Jakarta, Indonesia, with an augmentation process adopted to prevent overfitting. Different input image resolutions were also analytically utilized, i.e., 50 x 50, 100 × 100, and 150 × 150 pixels, and repeated at epoch 250. The results showed that the proposed AlexNet model had an accuracy validation value of 95%, due to its batch size parameter, learning rate, and splitting ratio of 8, 0.001, and 0.6:0.4, as well as an Optimizer SGD, training, and balanced data. Based on these findings, the model is considered promising for coffee granularity classification.

Keywords: Classification of images, *Deep Learning*, *Convolutional Neural Network*, Classification of *grinder* coffee *Alexnet*

Received: 15 March 2022 Revised: 30 May 2022 Accepted: 2 June 2022

1. Introduction

A plantation product enjoyed by Indonesians is coffee, which has a high economic value. It is also the main livelihood source of one and a half farmers in the country. Regarding the high demand for its products, various types of processed coffee have been identified, with much emphasis on the aspects of maintenance, pest and weed control, as well as harvesting. In addition, coffee is found to be processed into various types of drinks. This is consumed after the mashing of the beans, whose smoothening processes involve the utilization of a manual and automatic grinder. After grinding, the coffee is then processed into multiple drinks with distinguished flavours. Furthermore, the grinder is divided into 5 categories, namely fine, fine-medium, medium, medium-coarse, and coarse, whose outputs often lead to a different type of drink. According to this classification, the results of different grinders commonly produce distinct coffee tastes.

Convolutional Neural Network (CNN) helps to detect and distinguish the primary structural elements in images (Lu et al., 2020; Zhang et al., 2021). This is conducted by using the filter to adjust the whole image towards performing the pattern accordingly. In this process, CNN often captures these patterns at any specified location in the retina. The movement of this filter across the image is gradually conducted periodically, to determine the methods of matching the image patterns (Yin et al., 2020). CNN also contains several processing units, which are dependent on weight and self-study bias. This shows that each neuron obtains multiple inputs, which are multiplied by weights and biases, accompanied by an optional activation function (Lv et al., 2020; Sarıgül et al., 2019). The entire network also uses one distinguishable score function, from raw image pixels at one end to the class scores on the other. Based on the CNN Architecture, the input is an image, used to encode a specific property into structural design, leading to the highly-efficient implementation of advanced functions, which greatly reduces the network parameters (Unnikrishnan et al., 2018).

A successful approach to the development of intelligent systems is the ANN (Artificial Neural Networks), which was inspired by the performance patterns of human neuron networks and redeveloped in Deep Learning. These processes

* Corresponding author:
E-mail address: franky_leo70@student.esaunggul.ac.id

led to the development of the Convolutional Neural Network (CNN), whose learning process is often used for image processing and recognition. This is due to the grade and classification of the image analysis as a human function (Yalçın, 2021). Therefore, this study aims to classify coffee granularity using the CNN method. It also aims to evaluate the classification patterns of the coffee grinder's outputs, which are expected to produce processed commodities with high taste and excellent global competitive reputation. This is based on becoming a part of the Artificial Intelligence analysis (Esgario et al., 2020; Sarıgül et al., 2019). The grinder dataset was obtained from several coffee drink stores in Jakarta, Indonesia, with a pre-trained AlexNet model used in developing the classification system. The classification process was also analyzed regarding validity and accuracy, towards the application of the coffee grinder output. In addition, the results obtained are expected to accurately and efficiently classify the outputs of this grinder. Based on these conditions, the identification of problems is observed as follows:

1. Many results of the manual grinder have not been identified.
2. The slow process of identifying the grinder results.

This is because human errors are often observed in identifying the categories of coffee grinder results.

2. Related Works

Based on the applications for classifying coffee grinder, many similar reports have been carried out by analyzing different methods. As a reference, several journals have also been gradually analyzed in implementing the use of convolutional neural network methods. The following are the previous studies related to the implementation of these methods.

1) According to (Marcos et al., 2019), the CNN method was modelled and trained to identify rust infection signals. This was performed to approximately detect infected areas, with simple morphology used to improve the processes through high precision. It is also used to explore other CNN models, as well as the effects of layer sequences and their combinations with other postal processing techniques, towards rust infection detection and dice count coefficients.
2) According to (Putra, 2016), the classification of object imagery was quite reliable through the process of the convolutional neural network methods. This was based on the accurate values between 20-50%. In this condition, changes in the level of confusion did not affect these numeric results, indicating that CNN was relatively reliable against the conducted parameter adjustments. By using good and optimal information, the subset of the training data also produced a good classification.
3) The study of (Huang et al., 2019) explained that the CNN model was used to analytically classify good and bad seeds. By connecting the identification model to the webcam, good and bad seeds were instantly analyzed differently from the green coffee beans selected by the human eye. Using the object detection and image recognition technology, the involved time and labour costs were minimized. These results were used to help develop the coffee industry.
4) Based on (Tanuwijaya & Fatichah, 2020), the submission of a modified model was explained through Lite AlexNet, which was able to detect the availability of parking spaces using You Only Look Once (YOLO) V3. In this study, the proposed CNN Lite AlexNet model produced an accuracy rate above 92.33% with the best training time of 7 s to speed up the process.
5) According to (Prasetyo & Akardihas, 2019), the classification of batik motifs was analyzed using the CNN method of ResNet architecture, to recognize the product patterns. This showed that the augmentation of data training provided accuracy rates of 84.52 and 81.90% on ResNet-18 and ResNet-50, respectively. The use of datasets with rotation, random-erase, scale, and flip augmentation also provided accuracy elevations of 4.06, 9.38, 6.52, and 8.58%, respectively.
6) The study of (Akbar et al., 2021) analyzed Alexnet architecture using the PSO algorithm (Particle Swarm Optimization), for the classification of cervical cancer cell images. Basic optimizations also emphasized Epoch, Minibatch, Learning Rate, Image input resolution, and training-testing data ratio. Using the algorithm, accuracy was observed at 67% beyond the Resnet architecture standard of 6.22% (Akbar et al., 2021).

Based on these studies, the CNN method was applied and used with the AlexNet architecture (Gonzalez, 2007), to classify the quality of the coffee grinder results.

3. Methods

3.1. Dataset Setup

The analysis was initially carried out towards the preparation of a dataset, which was to be used in the training process. This set was obtained from the observations of several Premium Coffee Shops, which were part of the samples used to classify the grinder results. It also contained 5 categories (Parker, 2019), namely Coarse, Medium-Coarse, Medium, Medium-Fine, and Fine as shown in Table 1. In addition, the samples of these categories are presented in Figures 1-5.

Table 1. Comparison of the dataset values

Category	Size (mm)	Amount of data
Coarse	1.2-1.5	198
Medium Coarse	1.0-1.2	255
Medium	0.6-1.0	160
Medium-Fine	0.4-0.6	152
Fine	0.2-0.4	274
Total		1039

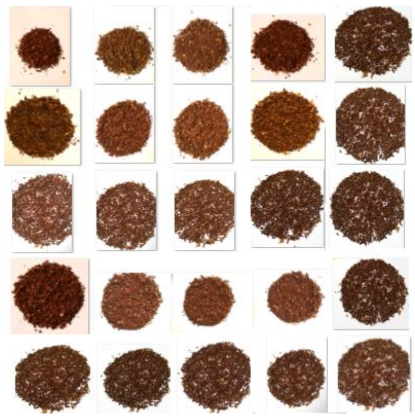

Figure 1. Samples of Coarse Category, size 1.2-1.5 mm

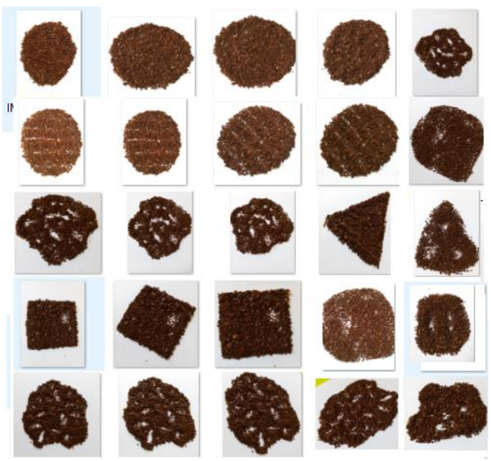

Figure 2. Samples of Medium-Coarse Category size 1.0-1.2 mm

Figure 3. Samples of Medium Category size 0.6-1.0 mm

Figure 4. Samples of Medium-Fine Category size 0.4-0.6 mm

Figure 5. Samples of Fine Category size 0.2-0.4 mm

3.2. Dataset Training

At this stage, the dataset was trained using the Convolutional Neural Network with the Python Programming Language, which is an interpreter that categorically processes program codes, compared to compilers. This was selected due to the provision of many libraries, which supported machine learning development. In addition, a Visual Studio Code and Jupyter Notebook software was used to operate Source Machine Learning and Application code, to obtain a high validation accuracy value.

3.3. Proposed Study

A CNN architectural model was proposed to classify the granularity of coffee grinders based on imagery. Figure 6 illustrates the experimental perception of this analysis, where image input sizes of 50 x 50, 100 x 100, and 150 x 150 pixels were trained using the AlexNet architecture. In this condition, 7 trial indicators were proposed, including Epoch, Batch size, Optimizer, Learning rate, Input Dimension, as well as Data Splitting and Balancing (Esgario et al., 2020).

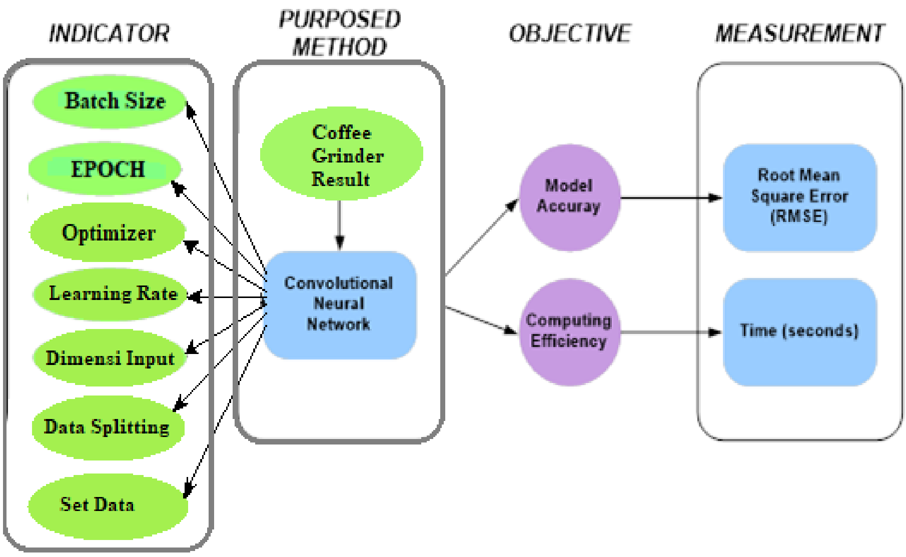

Figure 6. CNN Experimental Framework

4. Results and Discussion

To improve machine learning performance, optimization (Hyperparameter Tuning) was performed for the AlexNet's parameters, as shown in Table 2.

Table 2. CNN Testing Parameter

Parameter	Set Value				
Image Input Size	50 x 50	100 x 100		150 x 150	
Epoch	100	150		250	
Optimizer	ADAM	SGD		RmsProp	
Batch Size	8	16		32	
Learning Rate	0,001	0,01	0,1	0,5	0,0005
Data Split Ratio	0,5:0,5	0, 6:0,4	0,7:0,3	0,8:0,2	0,9:0,1
Data Balancing	Balanced			Imbalanced	

4.1. Experimental Results on Image Input Size

To determine accuracy, *training* was conducted with multiple image variations, including 50 x 50, 100 x 100, and 150 x 150 pixels, as shown in Table 3.

Table 3. Experiments of Image Resolution

No	Resolution	Accuracy (%)	CPU time (seconds)	Count of Params
1	50 x 50	80	1282	Total params: 24,322,629 Trainable params: 24,321,413 Non-trainable params: 1,216
2	100 x 100	87,5	3683	Total params: 29,565,509 Trainable params: 29,564,293 Non-trainable params: 1,216
3	150 x 150	91.25	6816	Total params: 46,342,725 Trainable params: 46,341,509 Non-trainable params: 1,216

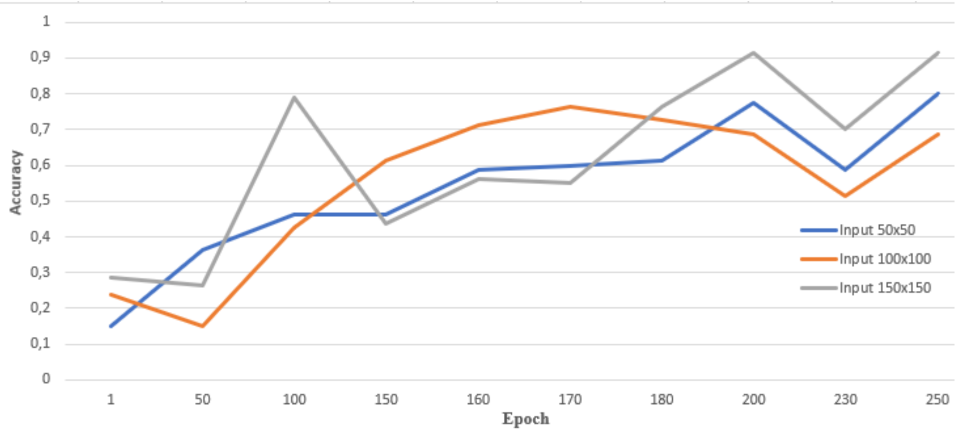

Figure 7. Graph Accuracy of Input Size

Figure 8. Graph Loss of Input Size

The results showed that the best accuracy value of 91.25% was observed at 150 x 150 pixels (Figures 7 and 8). Increasing the resolution value also affected the outputs of training time and parameters at 6,816 s and 46,341,509, respectively.

4.2. Experimental Results on Epoch

In the epoch stage, the entire dataset had undergone the training process on the Neural Network, until it is returned to the beginning for one round. This showed that more Epochs were needed for the system to improve weight and bias values. At this stage, the training process used quite a long time (Table 4 and Figure 9), with the highest epoch of 250 observed at 2,896 s. This was directly proportional to the accuracy increase of 90%. Based on these results, more epochs led to higher identification accuracy of the displayed image. However, very high epoch values were not allowed due to the restriction of longer training time. In this process, the number of datasets also affected the length of the training process.

Table 4. Comparison of Epoch Accuracy and Time

No	Epoch	Accuracy (%)	CPU time (seconds)	Count of Params
1	10	31,00	91	Total params: 29,565,509 Trainable params: 29,564,293 Non-trainable params: 1,216
2	20	31,00	183	
3	30	31,20	280	
4	40	31,25	382	
5	50	31,25	483	
6	60	31,25	583	
7	70	53,75	704	
8	80	83,75	834	
9	100	86,25	1088	
10	150	88,75	1687	
11	200	90,00	2287	
12	250	91,25	2896	

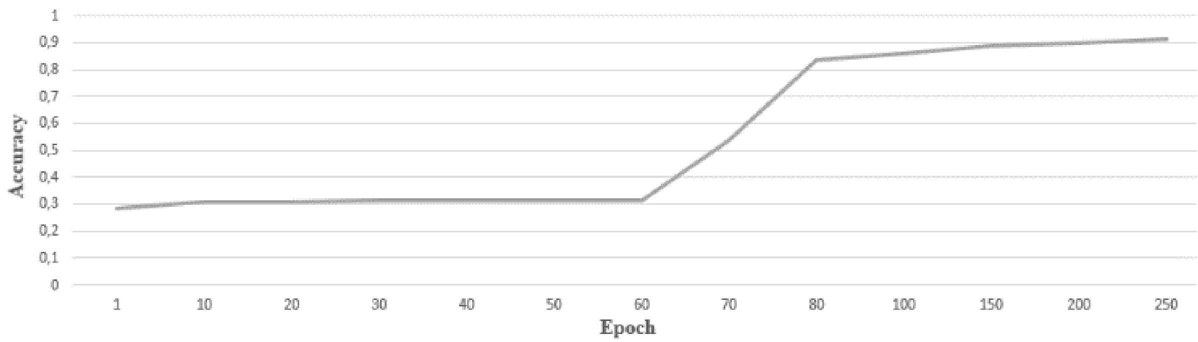

Figure 9. Graph Accuracy of Epoch

4.3. Experimental Results on Batch Size

To facilitate and speed up the training process, the data rate was divided per batch (Batch size), where various performances affected the obtained time and accuracy. In this case, 3 Batch size values were appropriately utilized, i.e., 8, 16, and 32 as shown in Table 5, as well as Figures 10 and 11.

Table 5. Experiment on Batch size

No	epoch	Accuracy (%) *Batch size* 8	*Batch size* 16	*Batch size* 32	Count of Params
1	10	25	25	25	Total params: 29,565,509 Trainable params: 29,564,293 Non-trainable params: 1,216
2	20	27,5	25	28,12	
3	30	27,5	25	28,12	
4	40	27,5	25	28,12	
5	50	32,5	26,25	28,12	
6	60	32,5	26,25	28,12	
7	70	37,5	38,75	45,63	
8	80	75	72,5	74,37	
9	100	82,5	78,75	83,75	
10	150	85	91,25	86,87	
11	200	87,5	91,25	89,38	
12	250	95	91,25	89,38	
CPU time (seconds)		1695	4226	7362	

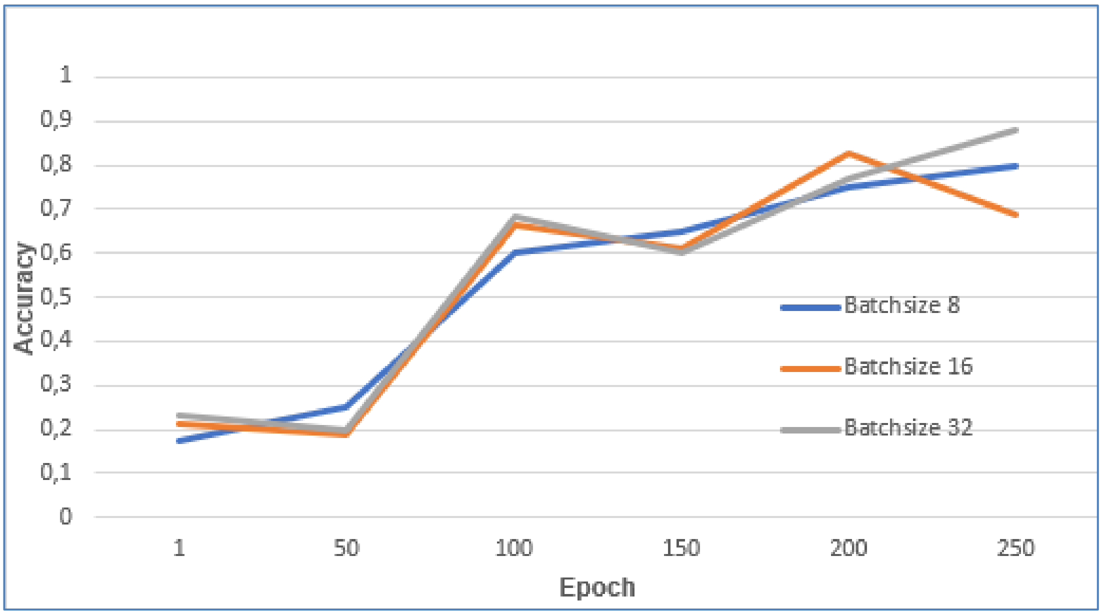

Figure 10. Graph Accuracy of Batch size

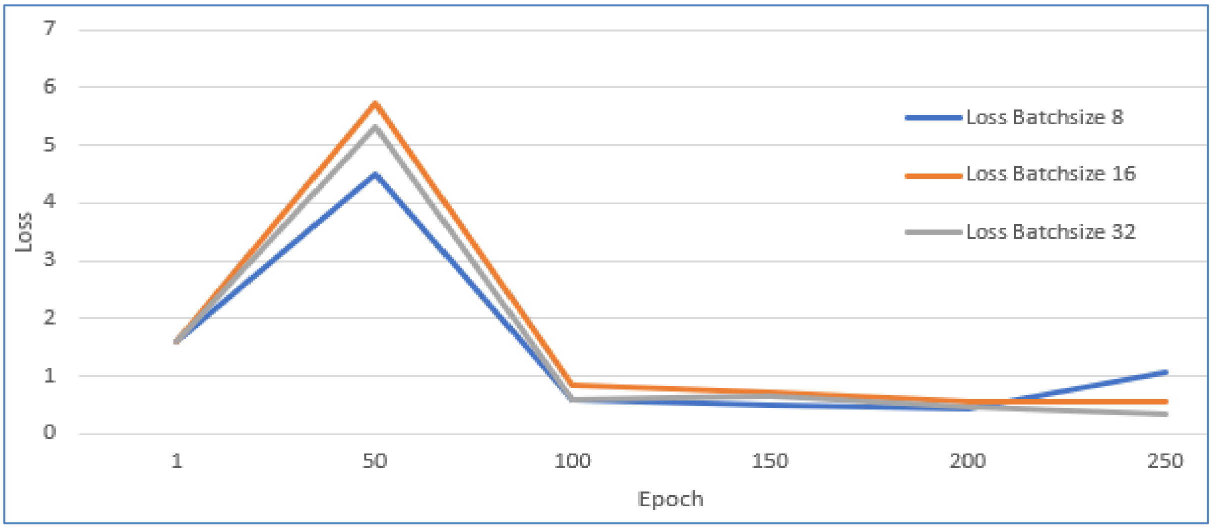

Figure 11. Graph Loss of Batch size

4.4. Experimental Results on Optimizer

Optimizers are often needed to determine, minimize, and maximize the optimal weight, errors, and accuracy, respectively. During the training process, the model parameters (weights) were transformed toward minimizing the loss function for accurate prediction. Meanwhile, the exact reasons many changes were still analytically uncertain led to the need for optimizers. In this process, 3 types of optimizers were used, namely SGD, Rmsprop, and Adam, as shown in Table 6, as well as Figures 12 and 13.

Table 6. Experiment on Optimizer

No	epoch	Accuracy (%)			CPU time (seconds)	Count of Params
		Adam	SGD	RmsProp		
1	10	0,2125	0,3125	0,275	91	Total params: 29,565,509 Trainable params: 29,564,293 Non-trainable params: 1,216
2	20	0,2625	0,3125	0,275	183	
3	30	0,2625	0,3125	0,275	280	
4	40	0,3125	0,3125	0,275	382	
5	50	0,3125	0,3125	0,3375	483	
6	60	0,35	0,3125	0,3375	583	
7	70	0,475	0,5375	0,5875	704	
8	80	0,6	0,8375	0,6375	834	
9	100	0,8	0,8625	0,825	1088	
10	150	0,8	0,8875	0,825	1687	
11	200	0,85	0,8875	0,825	2287	
12	250	0,875	0,9	0,875	2896	
				Total	11498	

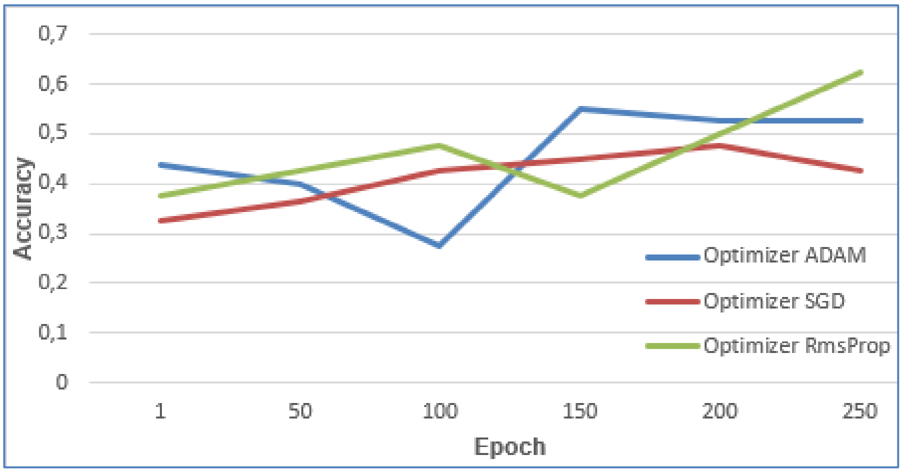

Figure 12. Graph Accuracy of Optimizer

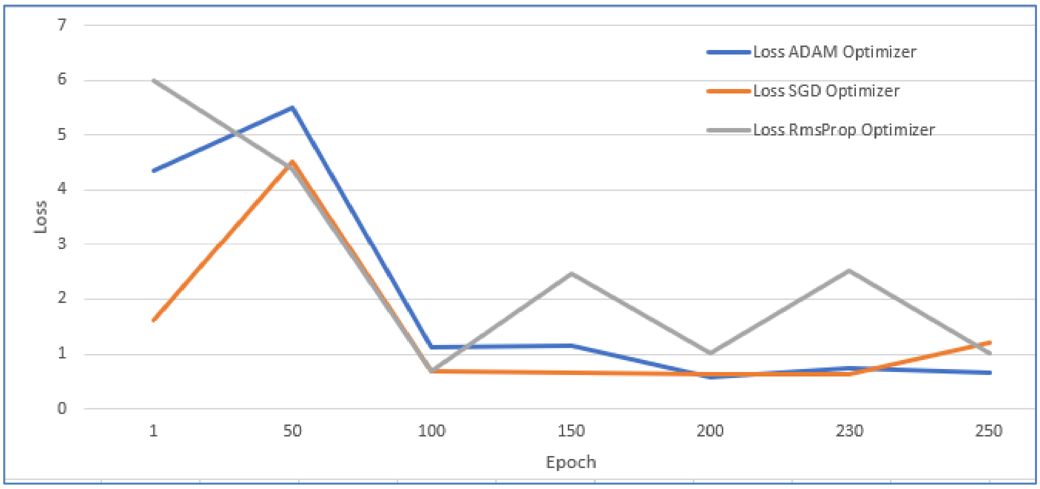

Figure 13. Graph Loss of Optimizer

4.5. Experimental Results on Learning Rate

The learning rate (LR) was used to calculate the weight correction value at the training time of 0 to 1. This indicated that greater LR led to a faster training process. However, the process exceeded the optimal state when the most minimal error was achieved, regarding the relatively larger learning rate value. This showed that LR affected the network accuracy of a system, with the configuration of the parameter observed at 0.0005, 0.001, 0.01, 0.1, and 0.5 (Table 7).

4.6. Experimental Results on Data Split Ratio

In analyzing a model, two dataset sharing were required, namely training and test data. In this condition, the training data were useful in conditioning the system toward classifying the coffee grinders' results. Meanwhile, the test data were used to test the accuracy of previously trained systems. Based on the results, the data-split ratios were observed as follows, 0.5:0.5, 0.6:0.4, 0.7:0.3, 0.8:0.2, 0.9:0.1, as shown in Table 8, as well as Figures 14 and 15.

Table 7. Experiments of Learning Rate

No	Minibatch	Vall Accuracy (%)	Learning Rate	CPU time (seconds)
1	8	90	0,0005	2218
2	16	90	0,0005	3404
3	32	90	0,0005	5684
4	8	95	0,001	1695
5	16	91,25	0,001	4226
6	32	89,38	0,001	7362
7	8	87,5	0,01	1650
8	16	90	0,01	3304
9	32	90	0,01	5790
10	8	42,5	0,1	2300
11	16	83,75	0,1	3917
12	32	89,38	0,1	5671
13	8	45	0,5	1603
14	16	32,5	0,5	2737
15	32	26,25	0,5	4691

Table 8. Experiments of Data Split Ratio

No	Training-Testing Split Ratio	Layer	Accuracy (%)	CPU time (seconds)
1	0,5:0,5	trimmed (balanced)	92,5	2761
2	0,6:0,4	trimmed (balanced)	95	4023
3	0,7:0,3	trimmed (balanced)	92,5	142 (early stopping)
4	0,8:0,2	trimmed (balanced)	91,25	2976
5	0,9:0,1	trimmed (balanced)	93,75	2662
6	0,5:0,5	Original (unbalanced)	87,5	4519
7	0,6:0,4	Original (unbalanced)	88,75	3999
8	0,7:0,3	Original (unbalanced)	91,25	2978
9	0,8:0,2	Original (unbalanced)	87,5	2896
10	0,9:0,1	Original (unbalanced)	88,75	2831

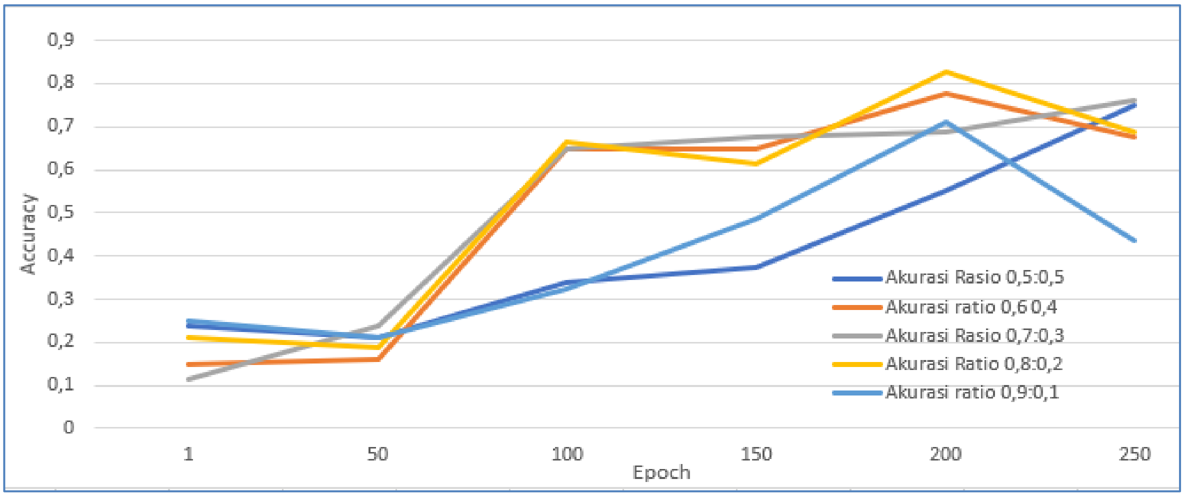

Figure 14. Graph Accuracy of Data Split Ratio

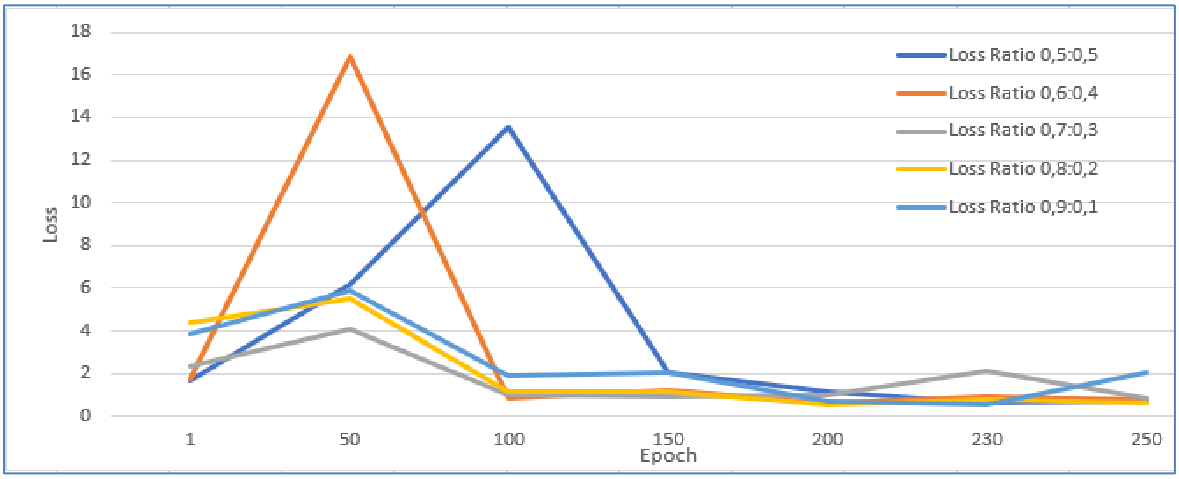

Figure 15. Graph Loss of Splitting Ratio

4.7. Experimental Results on Data Balancing.

With unbalanced classes, the classification of information was a major problem in the fields of machine learning and data mining. This was because the unbalanced condition portrayed a group with different data, compared to other classes. Therefore, the characteristics of data imbalance certainly affected the predictions produced by the algorithms, with subsequent analyses shown in Table 8.

5. Conclusions

Based on the results, parameter determination (Hyperparameter Tuning) affected the improvement of the CNN accuracy level in the classification of coffee grinder outputs. This proved that the image input dimensions of 150 x 150 pixels were the best, with an accuracy of 91.25%. In this condition, 250 Epoch parameters also increased the accuracy to approximately 91.25%. Furthermore, the utilization of an optimizer affected some improvements, with SGD observed to elevate accuracy by 90%. The Batch size parameter also improved the accuracy value by 8, leading to a total observation of 95%. For the learning rate, the use of 0.001 was able to increase accuracy by 95%. In addition, other adjusted parameters were the data splitting ratio and balanced accuracy settings of 0.6:0.4 and 95%, respectively. The balanced data also had the best computing time of 1,695 s. For future reports, other parameters such as variable convolution filters, activation function, number of neurons, etc, are needed for in-depth analytical performances. The test for training time and stability also needs to be futuristically carried out in more comprehensive studies.

References

Akbar, H., Anwar, N., Rohajawati, S., Yulfitri, A., & Kaurani, H. S. (2021). Optimizing AlexNet using Swarm Intelligence for Cervical Cancer Classification. *2021 International Symposium on Electronics and Smart Devices (ISESD)*, 1–6.

Esgario, J. G. M., Krohling, R. A., & Ventura, J. A. (2020). Deep learning for classification and severity estimation of coffee leaf biotic stress. *Computers and Electronics in Agriculture*, *169*, 105162.

Gonzalez, T. F. (2007). *Handbook of approximation algorithms and metaheuristics*. Chapman and Hall/CRC.

Huang, N.-F., Chou, D.-L., & Lee, C.-A. (2019). Real-time classification of green coffee beans by using a convolutional neural network. *2019 3rd International Conference on Imaging, Signal Processing and Communication (ICISPC)*, 107–111.

Lu, Z., Qin, Q., Shi, H., & Huang, H. (2020). SAR moving target imaging based on convolutional neural network. *Digital Signal Processing*, *106*, 102832.

Lv, M., Zhou, G., He, M., Chen, A., Zhang, W., & Hu, Y. (2020). Maize leaf disease identification based on feature enhancement and DMS-robust alexnet. *IEEE Access*, *8*, 57952–57966.

Marcos, A. P., Rodovalho, N. L. S., & Backes, A. R. (2019). Coffee leaf rust detection using convolutional neural network. *2019 XV Workshop de Visão Computacional (WVC)*, 38–42.

Parker, A. (2019). *What is Coarse Ground Coffee*. https://coffeedorks.com/coarse-ground-coffee/

Prasetyo, H., & Akardihas, B. A. P. (2019). Batik image retrieval using convolutional neural network. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, *17*(6), 3010–3018.

Putra, I. (2016). *Klasifikasi citra menggunakan convolutional neural network (CNN) pada caltech 101*. Institut Teknologi Sepuluh Nopember.

Sarıgül, M., Ozyildirim, B. M., & Avci, M. (2019). Differential convolutional neural network. *Neural Networks*, *116*, 279–287.

Tanuwijaya, E., & Fatichah, C. (2020). Modification of Alexnet Architecture for Detection of Car Parking Availability in Video Cctv. *Jurnal Ilmu Komputer Dan Informasi*, *13*(2), 47–55.

Unnikrishnan, A., Sowmya, V., & Soman, K. P. (2018). Deep AlexNet with reduced number of trainable parameters for satellite image classification. *Procedia Computer Science*, *143*, 931–938.

Yalcin, O. G. (2021). Deep learning and neural networks overview. In *Applied Neural Networks with TensorFlow 2* (pp. 57–80). Springer.

Yin, H., Gong, Y., & Qiu, G. (2020). Fast and efficient implementation of image filtering using a side window convolutional neural network. *Signal Processing*, *176*, 107717.

Zhang, X., Wu, F., & Li, Z. (2021). Application of convolutional neural network to traditional data. *Expert Systems with Applications*, *168*, 114185. https://doi.org/10.1016/j.eswa.2020.114185.