UDC 004
DOI: 10.15587/1729-4061.2023.289017

IDENTIFICATION OF CNN HYPER-PARAMETERS FOR TOBACCO LEAF QUALITY CLASSIFICATION ON NVIDIA JETSON NANO

Budhy Setiawan
Corresponding Author
Master of Electrical Engineering, Professor*
E-mail: budhy.setiawan@polinema.ac.id
Indrazno Siradjuddin
Master of Electrical Engineering, Doctor*
Arwin Datumaya Wahyudi Sumari
Master of Electrical Engineering, Doctor*
Widjanarko
Master of Mechanical Engineering, Lecture*
Eka Mandyatma
Master of Electrical Engineering, Lecture*
David Fydo Putradi
Master of Electrical Engineering, Student*
*Department of Electrical Engineering
State Polytechnic of Malang
Soekarno-Hatta str., 9, Malang, Indonesia, 65141

At the moment, there are some inaccuracies in manual classification for tobacco leaf quality selection, which are influenced by some factors, such as human fatigue or poor lighting. These cases lead to the need for another method that is more consistent, faster and reliable.

This research is an implementation of CNN (Convolutional Neural Network) in the classification of fresh tobacco leaves in terms of maturity grades. The primary objective is to develop an efficient CNN model capable to automize the classification of tobacco leaves into three maturity criteria: immature, mature, and old.

This methodology consists of some key factors, including color thresholding strategies to purge the noise from the background, Basic Image Manipulation approaches, the systemized screening of different input sizes, and CNN models to enhance the results.

The result of this research proves that the developed CNN model has 97.9 % accuracy achieved following 200 training sessions. The model is trained on a dataset comprising 1,249 fresh leaf photos, with a balanced 80:10:10 for train, validation and test ratio. However, the study emphasizes that the CNN model has successfully supported the tobacco leaf discrimination on a Jetson Nano Single-Board Computer with a Graphic Processing Unit (GPU).

The study extends beyond the mere theoretical contribution to practical applications in sorting "Gagang Rejeb Sidi" tobacco leaf, the highest quality tobacco variety in South Malang, East Java, Indonesia. Classification using a webcam as an input device shows the fastest processing time of 203.17 ms and the maximum is 1,363 ms.

This CNN model algorithm will be applied to a tobacco leaf selector machine, which has a high-speed conveyor and a three-position selector arm. The machine will be operated close to the field in post-harvest time under uniform lighting conditions.

Overall, the result of this research is highly relevant in terms of the short duration and accuracy for understanding the commodity classification. It provides a new angle toward speeding up the classification process and improving Indonesian tobacco quality

Keywords: fast tobacco leaves classification, Convolutional Neural Network, Nvidia Jetson Nano

Received date 15.09.2023
Accepted date 30.11.2023
Published date 29.12.2023

***How to Cite**: Setiawan, B., Siradjuddin, I., Sumari, A. D. W., Widjanarko, Mandayatma, E., Putradi, D. F. (2023). Identification of CNN hyper-parameter for tobacco leaf quality classification on nvidia Jetson Nano. Eastern-European Journal of Enterprise Technologies, 6 (2 (126)), 17–24. doi: https://doi.org/10.15587/1729-4061.2023.289017*

1. Introduction

Tobacco occupies a vital role in Indonesia's agricultural landscape by offering non-food commodity with high demand and price for both local and export needs. According to a report by the Directorate General of Customs of the Ministry of Finance of Indonesia, Tobacco Excise revenue in 2022 reached IDR 186.82 trillion. This figure increased by 15.54 % from the previous period. As the fifth largest global tobacco producer, Indonesia depends greatly on the East Java region, which accounts for about 44 % of the total national production [1]. Tobacco leaves have a complicated parameter as it depends upon the color, length and time in which leaves are harvested. It is important to classify leaves for cigarette production purposes. In which the best qualified leaf is the mature category. Since tobacco classification is still done manually, it is prone to human error due to the large number of leaves to sort through, which can cause fatigue, and also inconsistent lighting conditions.

The unique characteristics of tobacco leaves have led to ongoing attempts to automate the classification process. The research [2] used the Back Propagation Neural Network (BPNN) technique to classify objects based on their color, form, and texture features. It was evident that although BPNN achieved 77.50 % accuracy, it may not necessarily be the right technique for image processing.

That is quite an insight especially if one considers how complex are the image data structures and the size as well. Tobacco leaves have some specific features, and BPNNs do not always suit the tasks that have grid-like data.

This study then seeks to fill the gap by studying Gagang Rejeb Sidi tobacco, a premium variety from South Malang in East Java. The research objective is to categorize the leaves into three maturity classes: immature (less than 70 days), mature (70–100 days), and old (more than 110 days) [3].

In the context of cigarette manufacturing, the precise sorting of tobacco leaves holds significant importance. The suggestion from BPNN indicates the need for a neural network that is better suited to handle the complexities of this task. In present days, while CNNs can be complex with higher computational costs than BPNNs, current hardware development, including the device such as Jetson Nano, makes it possible to compress the execution time for these complicated algorithms. The integration of neural network powers with affordable hardware alternatives demonstrates that technology is ready to solve complicated issues related to, for example, the classification of tobacco leaves.

This highlights the significance of better neural network classification methods. Thus, research aimed at the improvement of maturation level-based and special-feature classified studies of tobacco leaves is definitely of high importance for upgrading the tobacco industry.

2. Literature review and problem statement

CNN has numerous applications in different domains. The use of IE-CNN comprised two internal-external features extracting modules for face recognition [4] area. Nevertheless, it is important to understand that the use of this method adds to the complexity of the training because of more parameters needed to be included in the model. The research paper [5] studies the performance of fingerprint scanning using the SGD optimizer for 64×64 and 48×48 inputs with 95.32 % accuracy. As well, [6] developed the model for sign language recognition using a CNN-based architecture such as 2 layers, 4 layers, 21 layers. The three models blended together showed a 98.6 % correct identification rate.

For a long time, people have tried to use Neural Networks to classify agricultural commodities. One study that used GRNN for tobacco leaf quality classification was [7], giving it around 93.5 % accuracy. Nevertheless, this results in a less than ideal scenario because the dataset only comprises 108 leaf samples. In addition, another report [8] used Raspberry Pi to measure mango size and ripeness. While classifying maturity based on color histograms and measuring size by a pixel-per-meters ratio revealed some classification limitations in this paper. A study by [9] used tobacco leaves photographs as datasets that usually fade, which give an unstable characteristic. For that, a DCNN-based model was supplied with both Apparent Features and Deep Features resulting in a superior test set accuracy of 98.75 %. Nonetheless, this remark is worth noting because the researchers only used static images instead of live data.

This study therefore takes a new approach after acknowledging the shortcomings of previous research attempts. A CNN model is run on Nvidia Jetson Nano, a single-board computer that is connected to a webcam serving as an input source and the classification of tobacco leaf. However, this decision was mainly due to impressive deep learning results achieved with the internal GPU of Jetson Nano compared to single-board PC competitors such as Raspberry PI without this feature. In addition, Jetson Nano is more economical as compared to Jetson TX [10]. These underlying challenges may be due to objective problems arising from the use of a limited dataset or certain fundamental constraints. It therefore leads to thinking of other ways of handling these problems. Therefore, the need for further study on this matter arises with lessons learned from previous experiences.

3. The aim and objectives of the study

The aim of the study is to develop a CNN model to classify fresh tobacco leaves according to their quality into three classes using Nvidia Jetson Nano.

To achieve this aim, the following objectives have been established:

– to make a CNN model using 100×100 input resolution;

– to make a CNN model using 224×224 input resolution;

– to evaluate CNN models ran on Nvidia Jetson Nano with a webcam as an input device.

4. Materials and methods

4. 1. Object and hypothesis of the study

The object of the study is to determine the maturity grades of fresh tobacco leaves. The main hypothesis of this research is that the CNN model can accurately and efficiently classify tobacco leaves into three maturity categories: immature, mature, and old. The work includes some assumptions, such as a perception that the color and texture of tobacco leaves change predictably as they mature. And this observation is learnable by the CNN's model. Therefore, some simplifications have been adopted in order to render the problem to be manageable. For example, the research focuses just on three stages of maturation categories and assumes uniform lightning for taking pictures of the leaves' dataset. These aspects of the research tend to offer a good starting point, to set up the basis for the CNN architecture followed by the analysis of its accuracies, high-speed classification time and finding the most appropriate input in between 100×100 and 224×224 pixels.

Convolutional Neural Network is an Artificial Neural Network method that is frequently used to analyze visual images. CNN architecture is divided into hardware and software parts [11–13]. CNN combines three basic architectures: local receptive fields, shared weight, which is a filter, and spatial subsampling, which is a pooling. Convolution is a matrix useful for the filter. There are several layers applied in CNN as a filter during the training process such as Convolutional Layer, Pooling Layer, Flatten, and Fully Connected Layer, as can be seen in Fig. 1.

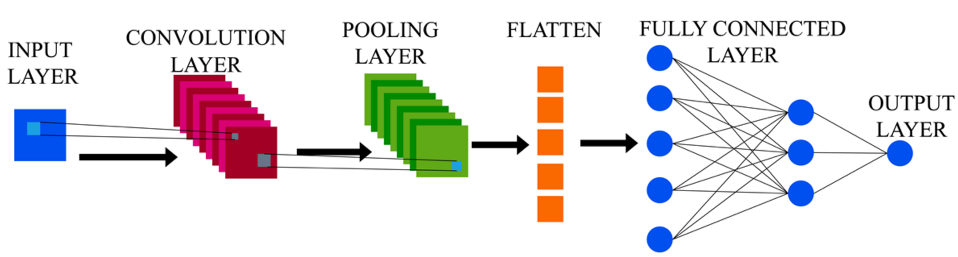

Fig. 1. Convolutional Neural Network architecture

In CNN, the input layer is the point of entry for raw data that could be an image or a feature vector. The next convolution layer involves filters that perform element multiplications to bring out spatial features in the input to enable the network to recognize patterns and structure. The pooling layer occurs after convolution where information is aggregated through max-pooling and other means of reduction in dimensions while maintaining critical features. However, the flatten operation makes the pooled output look like a 1D vector that is easily fed to the fully connected layers. Lastly, the output layer produces predictions or classifications, using softmax for multiclass classification and sigmoid for binary classification, thus making the CNN yield meaningful output to correspond with the input data.

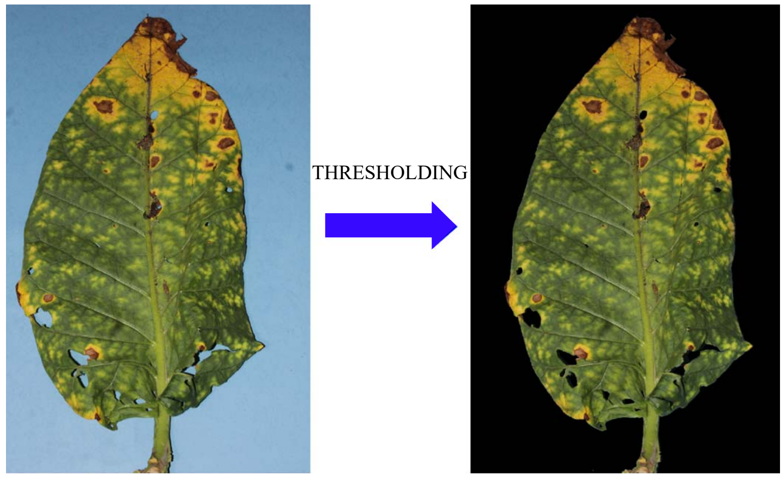

Fig. 3. Color thresholding

4. 2. Dataset acquisition

This research uses a dataset in the form of photos of individual tobacco leaves. Tobacco leaf is set under a Canon EOS 600D with a lens kit. We can see the positioning in Fig. 2. A portable LED light is used to provide uniform lighting. The picture is taken in JPG format.

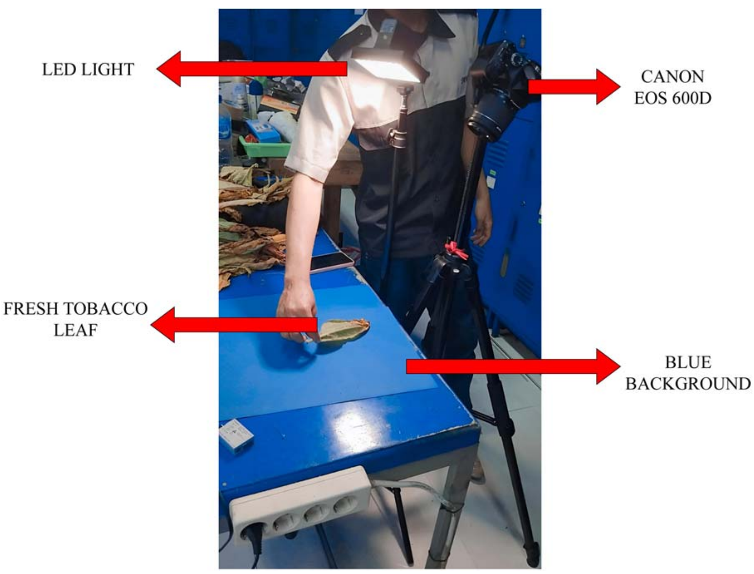

Fig. 2. Taking photos for the dataset

Since the photos have the same background shade, we can apply color thresholding to mask the background so the training can focus on the object, or in this case, the tobacco leaf. In Fig. 3, the blue background is removed and replaced with a uniform black (RGB 0, 0, 0). The colors removed are from RGB 50 ,50 ,90 to RGB 138, 255, 255.

Pixel-based image processing approach known as thresholding helps in the segmentation of an object and region of interest in a digital image. This process entails deciding on an explicit brightness level above which the pixels are classified as front and below which they are regarded as back. Binary classification based on pixel intensity enables well-defined object boundaries and makes the later image analysis and feature extraction easier to achieve, being an essential component for the whole image processing and computer vision applications.

4. 3. Convolutional Neural Network Architecture

The collected dataset consisting of 1,200 photos, 400 photos for each class, is divided into 80 % for training data, 10 % for testing data, and 10 % for validation data. Table 1 shows a sample for the classification of tobacco leaves of each class.

In the initialization stage, the program runs all the libraries needed. The main library used is Keras. Keras is a Python library that is used for deep learning purposes. Besides that, NumPy is used to run arithmetic functions, and Sklearn was also used to calculate the value of the convolution matrix. The dataset went through Basic Image Manipulation [1], which was used to enlarge the dataset. Basic Image Manipulation consists of rotation, flip, crop, and translation.

After that, the training data are put into CNN. In this study, two input sizes and several CNN architectures are tested to find the highest accuracy and fastest classification time. All architectures are trained on 200 epochs. This study uses a combination of 1 Convolution layer and 1 MaxPooling layer to 5 Convolution layers and 5 MaxPooling. The input sizes are 100×100 and 224×224 for memory efficiency when training the CNN. The specification of the fully connected Neural Network used in this study can be seen in Table 2.

Table 1

Tobacco leaf classes

		
Immature	Mature	Old

Table 2

CNN architecture specification

Kernel size	Filter	Strides	Activation	Pool Size	Pool strides
3	32	1	ReLU	3	2

All of the CNN tested in this study use the specification from Table 2. Kernel size 3 sets the size of the convolutional filter to 3×3 pixels, meaning each filter will examine a 3×3 region of the input data at a time. Filter 32 indicates that there will be 32 filters in this convolutional layer. These filters are small grids that slide over the input data to detect patterns. Strides determine how much the filter shifts when it convolves the input. A stride of 1 means the filter moves one pixel at a time. The Rectified Linear Unit (ReLU) activation function is applied after the convolution operation. ReLU is a common choice for activation functions in deep neural networks. Max-pooling is a down-sampling operation that reduces the spatial dimensions of the data. Here, a 3×3 pooling window is used to compute the maximum value within each 3×3 region. The pooling window moves with a stride of 2, which means it shifts by 2 pixels at a time.

4. 4. Model Evaluation

From 10 models tested on Nvidia Jetson Nano using a webcam, we choose the best model of each input and evaluate their performances. Accuracy, precision, and recall are used to evaluate the model. Accuracy refers to the proportion of the correct predictions that include both True Positive and True Negative and can be calculated using:

$$accuracy = \frac{TP+TN}{TP+TH+FP+FN}. \qquad (1)$$

Precision is defined as a measure of proportionate true positives made among all positive predictions by a model as an indicator of the potential to prevent false positives and can be calculated using:

$$precision = \frac{TP}{TP+FP}. \qquad (2)$$

Recall, also known as sensitivity or true positive rate, can be calculated using:

$$recall = \frac{TP}{TP+FN}, \qquad (3)$$

measures the proportion of actual positive cases correctly identified by a model.

5. Results of Convolutional Neural Network for tobacco leaf classification

5. 1. Test Result of 100×100

After 200 iterations of training, the models resulted training and testing accuracy. The 5 models tested have accuracy, which can be seen in Table 3.

Table 3 is the accuracy performance of the Convolutional Neural Network toward different parameters in training. The accuracy plot of the model with 4 Convolution and 4 MaxPooling is shown in Fig. 4.

Table 4 is the size of the resulting model of the Convolutional Neural Network toward different parameters in training. The models were then tested using Nvidia Jetson Nano implanted with the models. Table 5 provides the classification time required for each model.

Table 3

100x100 Training result

Input Size	Architecture	Training accuracy	Testing accuracy
100×100×3	1 Convolutional and 1 MaxPooling	0.918	0.916
	2 Convolutional and 2 MaxPooling	0.897	0.968
	3 Convolutional and 3 MaxPooling	0.866	0.935
	4 Convolutional and 4 MaxPooling	0.959	1.000
	5 Convolutional and 5 MaxPooling	0.742	0.677

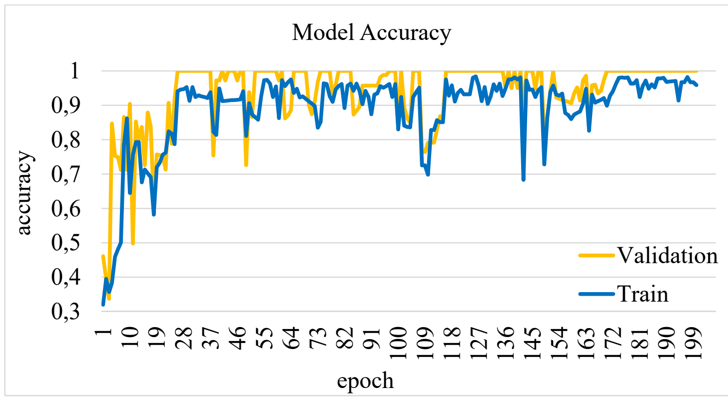

Fig. 4. Model accuracy for 4 Convolutional and 4 MaxPooling CNN with 100×100 input size

Table 4

100×100 Training result

Architecture	H5 File Size (KB)
1 Convolutional and 1 MaxPooling	462,602
2 Convolutional and 2 MaxPooling	112,318
3 Convolutional and 3 MaxPooling	57,274
4 Convolutional and 4 MaxPooling	48,855
5 Convolutional and 5 MaxPooling	80,069

Table 5

Classification time on Nvidia Jetson Nano of 100×100 input size

Input Size	Architecture	Minimum (ms)	Maximum (ms)	Average (ms)
100×100	1 Convolutional and 1 MaxPooling	270.01	896.06	283.79
	2 Convolutional and 2 MaxPooling	215.71	486.42	231.37
	3 Convolutional and 3 MaxPooling	203.14	549.40	221.53
	4 Convolutional and 4 MaxPooling	206.55	507.53	218.14
	5 Convolutional and 5 MaxPooling	212.51	612.40	225.06

Table 5 is the minimum, maximum, and average classification time of the CNN model embedded on Jetson Nano.

5. 2. Test Result of 224×224

From the training, we can get training and testing accuracy as stated in Table 6. These are the theoretical accuracy or on paper, but not the real performance when tested in real time.

Table 6

224×224 Training result

Input size	Architecture	Training accuracy	Testing accuracy
224×224×3	1 Convolutional and 1 MaxPooling	0.969	1.000
	2 Convolutional and 2 MaxPooling	0.948	0.961
	3 Convolutional and 3 MaxPooling	0.969	1.000
	4 Convolutional and 4 MaxPooling	0.979	1.000
	5 Convolutional and 5 MaxPooling	0.824	0.871

Table 6 is the accuracy performance of the Convolutional Neural Network toward different parameters in training. From the training process, the models are saved in H5 format. Their sizes are shown in Table 7. Fig. 5 shows the accuracy of the best model.

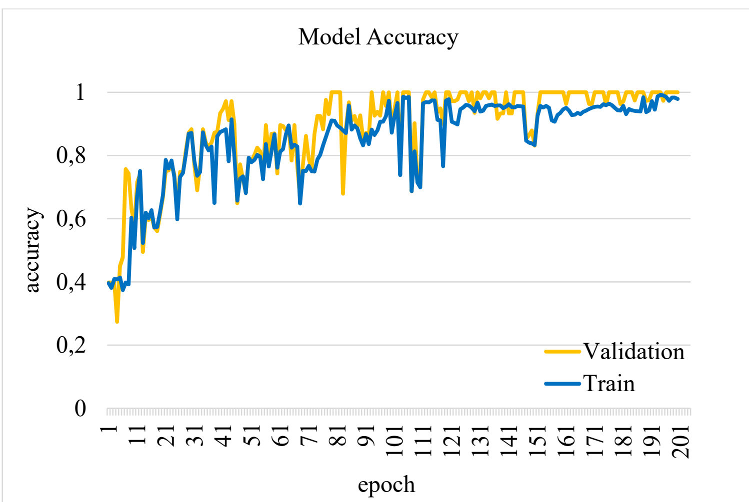

Fig. 5. Model accuracy for 4 Convolutional and 4 MaxPooling CNN with 224×224 input size

Table 7

224×224 Training result

Architecture	H5 File Size (KB)
1 Convolutional and 1 MaxPooling	2,367,242
2 Convolutional and 2 MaxPooling	1,863,219
3 Convolutional and 3 MaxPooling	281,887
4 Convolutional and 4 MaxPooling	261,975
5 Convolutional and 5 MaxPooling	257,269

Table 7 is the size of the resulting model of the Convolutional Neural Network toward different parameters in training. The models obtained were then embedded to Nvidia Jetson Nano. We can get the classification time needed for each model in Table 8.

Table 8

Classification time on Nvidia Jetson Nano of 224×224 input size

Input Size	Architecture	Minimum (ms)	Maximum (ms)	Average (ms)
100×100	1 Convolutional and 1 MaxPooling	316.50	1,363.52	358.71
	2 Convolutional and 2 MaxPooling	285.23	1,193.87	325.37
	3 Convolutional and 3 MaxPooling	274.65	914.51	296.34
	4 Convolutional and 4 MaxPooling	261.81	723.40	231.37
	5 Convolutional and 5 MaxPooling	270.01	896.06	283.79

Table 8 is the minimum, maximum, and average classification time of the CNN model embedded on Jetson Nano.

5. 3. Test Result on Nvidia Jetson Nano

The best architectures trained are put to test. The test consists of classifying the tobacco leaves using Nvidia Jetson Nano with a webcam as an input. The result of the classifications can be seen in Fig. 6.

From the confusion matrix, we can extract the True Positive, True Negative, False Positive, and False Negative from each class, as shown in Table 9.

Table 9

Performance metrics table for Nvidia Jetson Nano testing using 100×100 input on CNN model with 4 Convolutional and 4 MaxPooling

Class	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
Mature	22	30	6	2
Immature	15	31	3	1
Old	16	29	0	5

After getting the necessary metrics, we can evaluate the model tested on Nvidia Jetson Nano using accuracy, recall, and precision metrics, which can be seen in Table 10.

Table 10

Performance of Each Class on 100×100 model

Metric	Mature	Immature	Old	Micro average
Accuracy		–		0.8688
Precision	0.7857	0.8823	1	0.854
Recall	0.9166	0.9375	0.7619	0.868

Table 10 demonstrates that the model achieves greater than 85 % on each of the three metrics of recall, precision, and accuracy. This indicates that even though the outcome is not as good as during training, it is still trustworthy.

After the testing of the 100×100 input model is done, we move on to the testing of the 224×224 input model. The model is able to classify tobacco leaves in real time, as shown in Fig. 7.

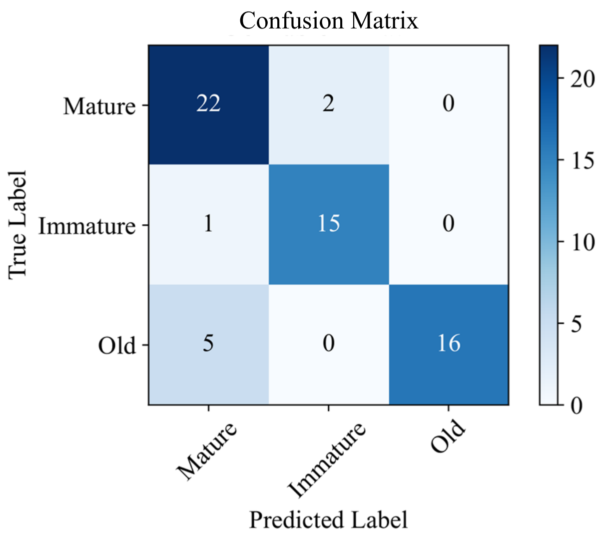

Fig. 6. Test result on Nvidia Jetson Nano using 100×100 input on CNN model with 4 Convolutional and 4 MaxPooling

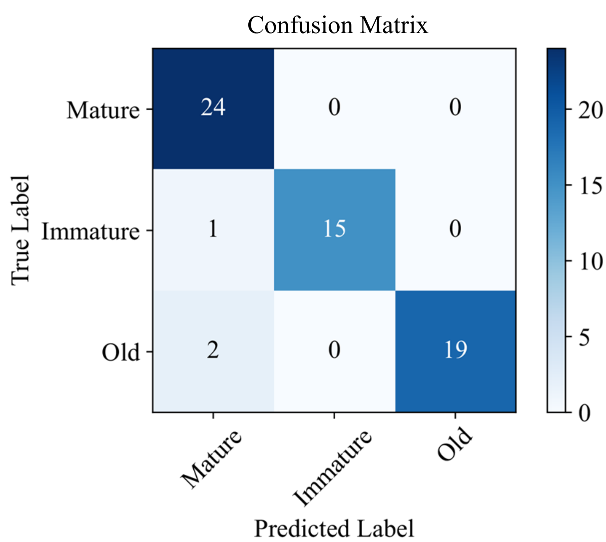

Fig. 7. Test result on Nvidia Jetson Nano using 224×224 input on CNN model with 4 Convolutional and 4 MaxPooling

From the confusion matrix, we can extract the True Positive, True Negative, False Positive, and False Negative from each class, as shown in Table 11.

Table 11

Performance metrics table for Nvidia Jetson Nano testing using 224×224 input on CNN model with 4 Convolutional and 4 MaxPooling

Class	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
Mature	24	45	3	3
Immature	15	48	1	1
Old	19	46	1	2

After getting the necessary metrics, we can evaluate the model tested on Nvidia Jetson Nano using accuracy, recall, and precision metrics, which can be seen in Table 12.

Table 12

Performance of Each Class on 224×224 model

Metric	Mature	Immature	Old	Average
Accuracy		–		0.947
Precision	0.889	0.937	0.950	0.920
Recall	0.889	0.937	0.904	0.906

Table 12 shows that the model obtained more than 90 % on all three metrics: accuracy, precision, and recall. This result means even though the result is not as good as in the training, it is still reliable.

6. Discussion of the results of Convolutional Neural Network for tobacco leaf classification

In Table 3, it is evident that the model with 5 Convolution layers and 5 MaxPooling yielded the poorest testing and training accuracy, measuring at 0.677 and 0.742, respectively. A similar trend is observed in Table 6 where training and testing accuracy was registered at 0.824 and 0.871. The result shows that the model using more Convolutional layers and MaxPooling layers tends to give worse results because even though the model has more capacity to learn intricate details and pattern in the training data, which is beneficial to some extent, it can also make the model more prone to overfitting, especially with the model that does not have a sufficient dataset. Overfitting occurs when the model learns to perform well on the training data but fails to generalize unseen data.

The model's accuracy as shown by its training and validation curve in Fig. 4 seems to increase along the epochs. However, our model is not static; instead, it oscillates frequently until up to the 175th epoch. The accuracy curve is shown in Fig. 5. The model oscillates larger at the start, but eventually approaches about one by 125th epoch. At the 150th epoch, there is the last big fluctuation on the model's path of convergence.

Looking at Tables 4, 7, it is apparent that architectures with 4 Convolutional layers and 4 MaxPooling layers generate smaller models regardless of the 100×100 input or 224×224 input. This finding is consistent with the information presented in Tables 5, 8, which illustrate that these models take the least amount of time for classification in Nvidia Jetson Nano, at an average of 221.15 ms for 100×100 and 231.37 ms for 224×224.

After testing the best models (with 4 Convolutional layers and 4 MaxPooling layers) for both 100×100 and 224×224 inputs on Nvidia Jetson Nano, the results are presented in Fig. 6, 7, along with the corresponding data in Tables 9, 11. These tables allow us to calculate metrics for both models.

From Tables 10, 12, we can analyze the performance of both models. Regarding the main metric of accuracy, Model 224×224 outperforms model 100×100 with an accuracy of 94.7 % against 86.88 % for model 100×100. Finally, precision values for Model 224×224 are high compared to that of the 100×100 Model in all the three classes (Mature, Immature, and Old). Precision refers to the percentage of instances classified into a particular class that actually belong to that class. The precision values for the 224×224 input model (mature=0.889, immature=0.937, old=0.950) are consistently higher than for Model 1 (mature=0.7857, immature=0.8823, old=1). This means Model with 224×224 reduces the risk of false positives. However, recall that measures how well a model captures all instances of a particular class, also shows increased performance with 224×224 input. The model's recall values (mature=0.889, immature=0.937 and old=0.904) are higher than those of model 100×100 (mature=0.91). The 224×224 model offers a better option for achieving optimal precision and recall trade-off. In addition, the micro-average precision and recall values present a unified representation of the accuracy of the model for every class, and once more, 224×224 input beats 100×100 input. The micro-average precision for Model 224×224 is 92 %; whereas, it is 85.4 % for Model 100×100 input. This means that Model with 224×224 is better at minimizing false positives. Additionally, recall – a measure of the model's competence in capturing all instances of a particular class – also shows improvement for the 224×224 input. Its recall figures (mature=0.889, immature=0.937, old=0.904) are greater than those of Model 1 (mature=0.9166, immature=0.9). Balancing on the precision and recall trade-off, the 224×224 model improved the precision and recall of these improvements. In addition, micro-average precision and recall values give an overall performance of a model in relation to all classes, and here too 224×224 input takes the lead as compared to 100×100 input. The micro average precision is 92 % for the model with the input of 224×224 and 85.4 % with the input of 100×100. Model 224×224's micro-average recall is 90.6 %, compared to Model 100×100's 86.8 %. These metrics illustrate that Model 224×224 is the more robust performer across all classes, and its superiority extends to the overall aggregated evaluation. The models obtained from this study generate a better accuracy.

Although this study has given satisfying results, it is still not perfect. The biggest factors of misclassifications in this study are the lack of a dataset, and inconsistent lighting conditions when testing using a webcam and Nvidia Jetson Nano. This can be overcome by collecting a larger and more diverse dataset. Better testing environment can also improve the result.

From the results of the tables above, Convolutional Neural Network can classify the quality of tobacco leaves with good accuracy and low classification time, which satisfies the needs of the industry to reduce time spent and increase production. This study suggests that Nvidia Jetson Nano can run a Convolutional Neural Network classifier model with enough speed to control an electric motor for further research in making a tobacco leaves selection machine.

7. Conclusions

1. The best model from the 100×100 input model uses 4 Convolution layers and 4 MaxPooling, which resulted in 0.959 and 1.000 in training and testing accuracy.

2. The best model from the 224×224 input model uses 4 Convolution layers and 4 MaxPooling, which resulted in 0.979 and 1.000 in training and testing accuracy.

3. Nvidia Jetson Nano can run CNN models smoothly and quickly with an average classification time of 267.513 ms. The real time accuracy for the 100×100 model is 0.8688 and 0.947 for 224×224 input. This means the best model tested only has a difference of 3.2 **%** from the simulation testing accuracy.

Conflict of interest

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper.

Financing

This research has been carried out under the support of Higher Education, the Ministry of Education and Culture of Indonesia in 2023, through the Industrial Cooperation Research scheme, which was funded by the internal independent budget of Politeknik Negeri Malang.

Data availability

Data will be made available on reasonable request.

Use of artificial intelligence

The authors have used artificial intelligence technologies within acceptable limits to provide their own verified data, which is described in the research methodology section.

References

1. Susanti, A., Waryanto, B. (Eds.) (2018). Agricultural Statistic 2018. Ministry of Agriculture Republic of Indonesia. Center for Agriculture Data and Information System.
2. Sari, Y., Pramunendar, R. A. (2017). Classification Quality of Tobacco Leaves as Cigarette Raw Material Based on Artificial Neural Networks. International Journal of Computer Trends and Technology, 50 (3), 147–150. doi: https://doi.org/10.14445/22312803/ijctt-v50p126
3. McMurtrey, J. E. (2023). tobacco. Encyclopedia Britannica. Available at: https://www.britannica.com/plant/common-tobacco
4. Song, A.-P., Hu, Q., Ding, X.-H., Di, X.-Y., Song, Z.-H. (2020). Similar Face Recognition Using the IE-CNN Model. IEEE Access, 8, 45244–45253. doi: https://doi.org/10.1109/access.2020.2978938
5. Harivanto, Sudiro, S. A., Kusuma, T. M., Madenda, S., Rere, L. M. R. (2020). Detection of Fingerprint Authenticity Based on Deep Learning Using Image Pixel Value. 2020 Fifth International Conference on Informatics and Computing (ICIC). doi: https://doi.org/10.1109/icic50835.2020.9288589
6. Suardi, C., Handayani, A. N., Asmara, R. A., Wibawa, A. P., Hayati, L. N., Azis, H. (2021). Design of Sign Language Recognition Using E-CNN. 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT). doi: https://doi.org/10.1109/eiconcit50028.2021.9431877

7. Liu, J., Shen, J., Shen, Z., Liu, R. (2012). Grading tobacco leaves based on image processing and generalized regression neural network. 2012 IEEE International Conference on Intelligent Control, Automatic Detection and High-End Equipment. doi: https://doi.org/10.1109/icade.2012.6330105
8. Mustaffa, I. B., Khairul, S. F. B. M. (2017). Identification of fruit size and maturity through fruit images using OpenCV-Python and Rasberry Pi. 2017 International Conference on Robotics, Automation and Sciences (ICORAS). doi: https://doi.org/10.1109/icoras.2017.8308068
9. Luo, H., Zhang, C. (2018). Features Representation for Flue-cured Tobacco Grading Based on Transfer Learning to Hard Sample. 2018 14th IEEE International Conference on Signal Processing (ICSP). doi: https://doi.org/10.1109/icsp.2018.8652385
10. Suzen, A. A., Duman, B., Sen, B. (2020). Benchmark Analysis of Jetson TX2, Jetson Nano and Raspberry PI using Deep-CNN. 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA). doi: https://doi.org/10.1109/hora49412.2020.9152915
11. Valueva, M. V., Nagornov, N. N., Lyakhov, P. A., Valuev, G. V., Chervyakov, N. I. (2020). Application of the residue number system to reduce hardware costs of the convolutional neural network implementation. Mathematics and Computers in Simulation, 177, 232–243. doi: https://doi.org/10.1016/j.matcom.2020.04.031
12. Pandian, J., Kumar, V., Geman, O., Hnatiuc, M., Arif, M., Kanchanadevi, K. (2022). Plant Disease Detection Using Deep Convolutional Neural Network. Applied Sciences, 12 (14), 6982. doi: https://doi.org/10.3390/app12146982
13. Lu, M., Wang, C., Wu, W., Zhu, D., Zhou, Q., Wang, Z. et al. (2023). Intelligent Grading of Tobacco Leaves Using an Improved Bilinear Convolutional Neural Network. IEEE Access, 11, 68153–68170. doi: https://doi.org/10.1109/access.2023.3292340