Journal of Theoretical and Applied Information Technology
15th June 2023. Vol.101. No 11

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MATURITY CLASSIFICATION OF TOMATOES USING CONVOLUTIONAL NEURAL NETWORK

[1]OCTAVIANUS JAMES MARSHAL, [2]RIYANTO JAYADI

[1, 2] Information Systems Management Department, BINUS Graduate Program – Master Of Information Systems Management, Bina Nusantara University, Jakarta Indonesia 11480

E-mail: [1]octavianus.marshal@binus.ac.id, [2]riyanto.jayadi@binus.edu

ABSTRACT

Tomato is a fruit used daily that requires good quality, and quality is also an essential factor in sales percentage. In the market. Currently, the selection of quality tomatoes is still made mainly by humans, with several areas for improvement, such as the accuracy and consistency of the results obtained due to limited human perception. With ever-improving technology, it is now possible to train computers to classify images based on specific characteristics. This study proposes a classification model to classify tomato images using Convolutional Neural Networks (CNN). A total of 300 pictures of tomatoes have been selected from 480 pictures taken using a smartphone camera, and these images will be divided into three classes, unripe, ripe and rotten. Each class consists of 100 images and will be divided into 70% as training data, 15% as validation data and 15% as test data. In this study, we compared the accuracy of the VGG19 fine-tune and unblock layer models. And will compare the kernel used in the VGG19 model to determine the impact of the kernel on the classification accuracy that increases the model kernel parameters. From this study, CNN can see that the parameters used can affect accuracy.

Keywords: *Classification, Neural Network, Tomato, Maturity*

1. INTRODUCTION

The agricultural industry is currently multiplying. Indonesia is an agricultural country that has great potential in the agricultural industry. Indonesia has a tropical climate with relatively high rainfall and fertile soil for growing various plants. One of the many types of fruits and vegetables that have the potential to grow in Indonesia is tomatoes[1]. Tomatoes have many uses, including as a daily food ingredient, as vegetables, seasonings, coloring agents, cosmetics, and raw materials for food processing such as sauces, drinks, and others. Therefore, tomatoes are a vegetable and fruit with high economic value[2]. Tomatoes also have many benefits because they are rich in nutrients, vitamins, minerals, fiber and antioxidants, which are very beneficial for the health of the human body. In addition, there is data that tomatoes are one of the most traded products internationally, which means tomatoes are one of the goods needed in everyday life.

The need for tomatoes in Indonesia is relatively high compared to the number of tomatoes produced in Indonesia. Based on data from the Badan Pusat Statistik (BPS) for 2021, there is data that tomatoes are a vegetable and fruit crop with the potential to be placed sixth after several other vegetables and fruits such as shallots, potatoes, cabbage, mushrooms and chilies. Tomatoes with a production contribution of 1,114,399 tons which increased from the previous year of 29,406 tons or around 1.03%, and the largest tomato producer in Indonesia was West Java with a production of 299,267 tons, followed by North Sumatra Island with around 203,162 tons and West Sumatra Island amounted to 97,271 tons [3].

The increase in the need for tomatoes is felt by the people of Indonesia and throughout the world. Cause the increased demand triggers the cultivation of tomato plants to continue growing, along with the increasing demand for tomatoes with good ripe conditions. This condition also opens up opportunities to export fresh tomatoes. Several obstacles are often faced, namely the need for more availability of high quality, high production, good quality fruit and resistance to pests and diseases. Even though tomatoes are an agricultural product with high value, they have drawbacks such as being easily damaged, not being stored for a long time, and improper handling will accelerate the decline in quality, impacting nutrition and market value.

The process of selecting tomatoes by various companies and farmers is generally only done manually and involves humans as a determinant of whether the selected tomatoes' quality is the same. Manual identification of tomatoes has several drawbacks, such as requiring a relatively long time and the possibility of human error, which causes different perceptions of fruit quality and yield due to human visual limitations as well as these things can also result in fruit quality obtained from an inconsistent selection process and manual selection, which also requires a lot of time and effort.

Therefore, the problem of manual selection needs to be overcome by proposing a model that automatically recognizes tomatoes' degree of ripeness. Like the implementation of image processing, one of the techniques that can be used is the Deep Learning technique, which has an image processing method, namely the Convolutional Neural Network (CNN). CNN can be used to build architectural systems and perform image processing. CNN has the architecture of the human visual cortex to maximize the ability to manage image information. One of the CNN models often used, such VGGnet.VGG19 as part of VGGnet, will be used as a classification model for the degree of maturity of tomatoes in this study.

This research has the scope of the convolutional neural network method, then the use of the VGG19 architecture and tomatoes as research objects to be classified images. As well as this research aims to determine the use of the VGG-19 model in classifying the level of ripeness of tomatoes using several predetermined parameters. So with this research, CNN can be considered a method in selecting tomatoes based on the level of ripeness that will simplify and make human labor more efficient in selecting tomatoes.

2. LITERATURE REVIEW

Tomatoes will be classified into several classifications, such as unripe, ripe and rotten. The method that will be applied is deep learning using the convolutional neural network technique and the VGG-19 model. Classification is done based on optical surface features like color, texture etc.

2.1 Raw Materials

This study will be focused on the classification of tomatoes. Then the materials used are tomatoes, and the researcher will collect the tomatoes used as a dataset from traditional markets. The tomatoes purchased have the same variety. A total of 10 tomatoes have been collected and given the same treatment, namely washed, dried and prepared for data collection as data that tomatoes will use for classification.

2.2 Software Tools

This research will develop CNN using the Python programming language and uses Google Collab as the platform that the research will use to perform deep learning functions. In addition to developing it, several libraries are needed, such as Keras and tensor flow, which will help build Neural Networks. The library will provide features that this research will use to facilitate model development. And several libraries will be used to form histograms or diagrams that will help researchers determine the accuracy and accuracy of the VGG-19 model, such as pandas, plot, NumPy, and seaborn.

2.3 Convolutional Neural Network

Convolutional Neural Network (CNN) is a neural network whose primary input is images because CNN is devoted to processing grid-structured data. CNN also implements filters (features) engineered by hand in traditional algorithms. The main advantage is that CNN has also been developed from previous knowledge used to assist humanitarian efforts in feature design [4].

Journal of Theoretical and Applied Information Technology
15th June 2023. Vol.101. No 11

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

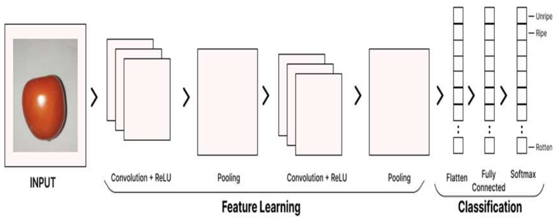

Figure 1 Convolutional Neural Network Architecture

And the CNN architecture will be divided into two parts, as shown in Figure 1, namely the Feature Extraction Layer and the Fully Connected Layer. The Feature Extraction Layer is the layer that changes from input to features that contain numbers indicating the image. This layer is divided into two parts, namely the *Convolutional Layer* and *Pooling Layer,* which in this layer will produce multidimensional array results that the next layer will process. The extraction results from the last pooling layer will go through a flattening process, converting the results into vectors. After that, the vector will be used as input to the second layer, the Fully Connected Layer. All neuron activity from the previous layer will be connected in this layer.

In the CNN structure, it is divided into several layers, including:

1) Convolution Layer

Layer convolution will utilize the kernel (filter) to process the image, where the kernel will be applied to the input image. As shown in Figure 2, one convolution process has several attributes, such as the green color block being the kernel above the input (blue block). The kernel will move from the top left to the bottom right, and a sum of the convolution results is entered into the feature map (red box).

The ability to extract features from the input image is one of the many advantages of the convolution layer. In addition, the dimensions of the layer determine the kernel convolution used so that the number of kernels in the convolution layer can be changed according to the input data. The extracted convolution layer will create a feature map from the input according to the spatial information contained in the input data, namely images [5].

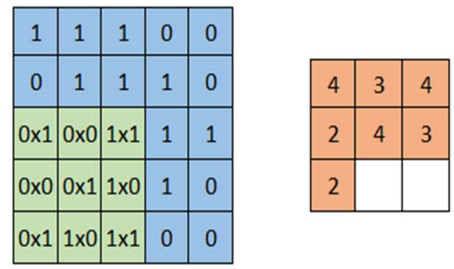

Figure 2 convolution

Several parameters can be changed in the Convolution layer: filter size, stride padding, and *activation function*. The function of stride is to determine the number of filter shifts where the smaller the steps, the more information is likely to be obtained, but it will require more processing than significant steps [6]. Padding or *zero padding* is a parameter that helps CNN determine the number of pixels containing zero values to be added on each side of the input. Ts convolution process produces *output* that will be used as input for the next layer [7]. In addition to calculating the dimensions of the feature map, the following calculations are performed, as shown in Figure 3.

$$\text{Output} = \frac{W+N+2P}{S} + 1$$

Figure 3 Feature Map

Output is a feature map obtained from the sum of Was input length/height, N as kernel length/height, P *zero padding*, and S as steps.

In addition, there is *Activation Function* which is used to set linear functions and help nerves to carry out more complicated tasks such as modeling complex data types such as images, video, audio, courses, text etc. There are also various activation functions, such as ReLu, Tanh, and Softmax [8].

2) **Pooling Layer**

After the convolution operation, it is usually through a pooling layer that will perform processes to reduce the dimensions and number of parameters to learn. Cause of that will shorten training time, require fewer data for maintenance, and combat overfitting. The pooling layer downsamples each feature map independently, reducing the height and width but keeping the depth intact. The most frequently used type of pooling in CNN is max pooling. Max pooling will extract maximum activation, while average pooling reduces activation weight by recording non-maximum activation [9]. As well as, the size of this window/block needs to be determined beforehand, which will reduce the size of the feature map and will only store essential and significant information in the later classification process. In addition, the commonly used pooling filter size is 2x2, and two strides as the pooling used. After that, max pooling will run on each data input slice. This shape will reduce the feature map from its original size. Figure 4 below shows an example of a Max Pooling operation.

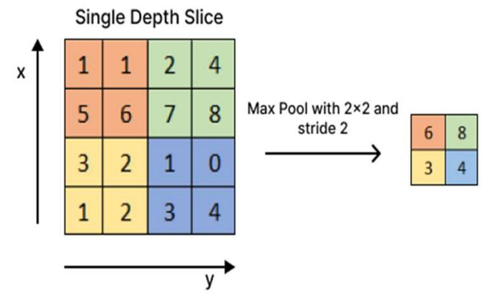

Figure 4 Max Pooling Layer

According to [10], A pooling layer is used on CNN to condense the image size such that a convolution layer can process it in the same steps as the pooling layer in question.

3) **Fully Connected Layer**

Layer Fully Connected Layer is the last layer, the connected center of all neural networks, and this layer aims to change the dimensions of the input data so that it can be classified. In addition, the neurons in the convolution layer must be converted to one dimension before proceeding to the fully connected layer. Because if it is not changed first, it can cause the data to lose its spatial information permanently, and the fully connected layer is at the end of the CNN architecture.

Furthermore, a fully connected layer is called a fully connected layer. So because the final feature map is still in the form of a filter matrix, it must be flattened before it can be used as input by the fully connected layer, where each neuron will be connected to every quantity and volume[11]

2.4 VGG-19

VGG-19 is an architectural model originating from the Convolutional Neural Network (CNN) and a variation of VGGNet. This model consists of 19 convolution layers, three fully connected layers, five pooling layers, and 1 SoftMax layer. And when training is carried out, this model also requires input in the form of an image with a pixel size of 224 x 224 pixels [12]. Figure 5 shows an architecture that can describe the VGG-19 model.

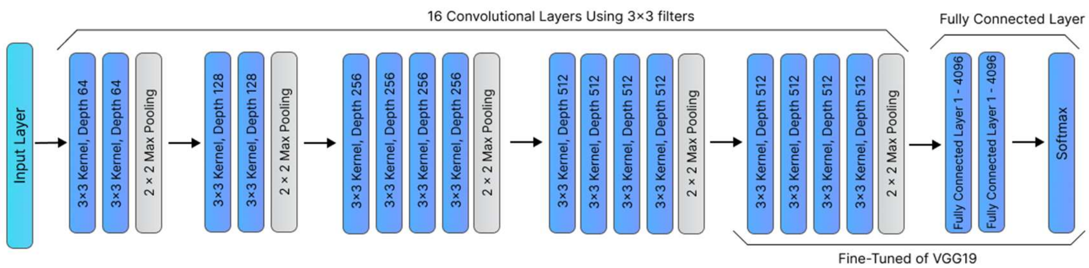

Figure 5 VGG-19 architecture

Based on Figure 4, the VGG-19 architecture begins with the input layer. The input layer can be an image processed through the second layer, namely the convolutional layers interconnected with the *max unification layer*. As well as this image input will pass through several convolutional layers with 3x3 filters with dimensions 64,128, 256, and 512. Then the input that has passed through several convolutional layers will go through flattening and will continue through 3 *Fully Connected Layers,* which consist of 2 layers of 4096-dimensional features and 1000-dimensional features and softmax activation functions.

2.5 Related Work

Author	Title	Summarize
Ninja Begum and Manuj Kumar Hazarika	Maturity detection of tomatoes using Transfer Learning	This study tested several transfer learning models in detecting the ripeness level of tomatoes, and this study used a transfer learning model with fine-tuning. This study also tested several parameters such as epoch, batch size, etc. The study found that fine-tuning VGG-19 got the best result[13].
Jose Luis Rojas-Aranda, Jose Ignacio Nunez-Varela, JC Cuevas-Tello, and Gabriela Rangel-Ramirez	Fruit Classification for Retail Stores Using Deep Learning	This research discusses fruit classification in retail stores using deep learning methods and combined with some improvisatio

		ns on MobileNetV2 with a single color, histogram and K-means. The results of this study using a single RGB color have the best accuracy among other improvisations[14].
Santi Kumari Behera , Amiya Kumar Rath, and Prabira Kumar Sethy	Classification of papaya fruit maturity status based on machine learning and transfer learning	approaches This study discusses the classification of papaya fruit using machine learning and transfer learning. The results of research using transfer learning and machine learning can be increased by the quantity of repetition of accuracy validation training.[15]
Dhiya Mahdi Asriny, Septia Rani, and Ahmad Fathan Hidayatullah	Orange Fruit Images Classification using Convolutional Neural Networks	This study discusses the classification of images on citrus fruits. This research was conducted on *activation functions* using the CNN model. This study

		found that the performance of ReLu activation was better than TanH.[16]
Tuti Purwaningsih, Imania Ayu Anjani, Pertiwi Bekti Utami	Implementation of Convolutional Neural Networks for Chili Classification	The categorization of chilies using a convolutional neural network is covered in this article. The study findings show that by utilizing CNN with specified parameters, training data can be predicted with an accuracy of 97.14% and test data with an accuracy of 80%[17].
Thanh-Hai Nguyen, Thanh-Nghia Nguyen and Ba-Viet Ngo	A VGG-19 Model with Transfer Learning and Image Segmentation for Classification of Tomato Leaf Disease	This study discusses the classification of Tomato Leaf Disease using the VGG-19 model with *Image Segmentation*. This study obtained results using the image segmentation method to increase accuracy and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

		speed in classifying compared to VGG-19 without image segmentation[18].
Rismiyati and Ardytha Luthfiarta	VGG16 Transfer Learning Architecture for Salak Fruit Quality Classification	This work aims to identify the quality of salak fruit quality using machine learning. One salak fruit is divided into two classes using one transfer learning technique, VGG16. With several stated parameters, the study findings showed a surprisingly high degree of accuracy [19].

Table 1 Related Work

3. METHODOLOGY

3.1 Data Collection

The researcher obtained 480 tomato images taken using a smartphone camera with a 64MP resolution. Images are obtained with room conditions that have non-uniform lighting. Lighting conditions do not refer to both in the sun at 10 to 3 pm. Pictures were taken directly from buying tomatoes. Approximately 20 images were taken two days apart, each shot over 24 days. Thus, the dataset was created from the images obtained by a predetermined method and consisted of unripe, ripe and rotten tomatoes. Of the 480 images taken to become the dataset used in this study, only 300 images will be used.

3.2 Data-Processing

The necessary processing is done on a laptop after the images from the smartphone camera have been transferred. Processing in this context refers to all preprocessing procedures and implementation models. Three hundred previously chosen photos will initially be used for processing, where the chosen images will first be cropped using the OS python library. Then the images will be resized using the OS python library with a size of 500 × 500. Second, the names of each classification will replace the original names, namely (Unripe, Ripe, and Rotten).

Additionally, the Red, Green, and Blue (RGB) color channels make up each pixel value in the tomato image. This pixel value contains details on the tomato's outside characteristics. Here is how to prepare a dataset: The first step is choosing which photographs will be utilized; after going through the phases listed above, the two images will be divided into three groups, unripe, ripe, and rotten, as shown in Figure 4. Each image is then displayed in Figure 4. Accordingly, each class is given a designation. The photos will be randomly separated into three groups for the study: training, testing, and validation data. These groups are named respectively as training data, testing data, and validation data, and they divide the data in the ratio 70:15:15.

Figure 4 maturity classes of tomatoes

3.3 Fine-Tuning

Fine-tuning is one of the training models that will use pre-trained models that have been turned off but not frozen. Even though the model has been paid for, the pre-trained model will later add to the convolution's last layer or retrain from scratch for new needs.

Furthermore, the last layer provides higher-level information. Thus, it is not frozen for fine-tuning because some frozen layers are nearest to the input layer and hold little information. As a result, the fine-tuning procedure will use the unique features offered at a layer higher than the base model. Next, the higher-order base model layers are not frozen, and the original base model layers are fixed as unable to pay to apply the fine-tuning principle. The model must then be recompiled to reflect the changes, and the training model must be restarted[20].

And the architecture that will be used in fine-tuned VGG19 is shown in Figure 4, where the architecture will start from the last convolution layer, namely 512 dimensions with a 3x3 kernel and will continue through the fully connected layer and will produce output. In addition, the use of fine-tuning will affect the total, and trainable parameters, which in fine-tuning, the total parameters are 20,552,771 and the trainable parameters are 528,387.

3.4 Hyper-Parameters

To help the model learn, several parameters known as hyper-parameters are employed in creating the VGG19. The hyper-parameters employed, such as kernel, number of epochs, batch sizes, and learning rates, fluctuate during the designated training models during development to attain the best classification accuracy in a model. Since this is each model's goal, it is vital to determine the ideal hyper-parameters.

Hyper-parameter values affect the model's performance, which helps it achieve the highest classification. Epoch is the number of training-related iterations performed over the entire dataset. With an Epoch value of 50, the model is on alert. The number of images from the training dataset is used in each iteration to correct gradient errors before updating the weights, known as the batch size. The sizes used in this investigation are 32 and 64. The batch size is determined during the training phase, and the CNN model works in batches. The architecture's performance can be maximized by achieving the best combination of hyper-parameters. Image input models will encounter problems with complex library image data generators in addition to reducing over- and under-fitting issues.

The input photos from the dataset will be rescaled and cropped by the image data generator in the VGG19 build model. The warning model will also include an optimizer named "adam" with a fixed learning rate of 0.0001 and a loss function called "categorical cross-entropy." Additionally, two kernels—3x3 and 5x5—will be utilized in this study, where the changes in the number of parameters will impact the accuracy of the model's output. Table 2 lists the hyper-parameters adjusted during training to maximize the graphical model.

Parameter	Value
Epoch	50
Batch-size	32 and 64
Learning-rate	0.0001
Loss function	Categorical_crossentropy
Kernel	3x3 and 5x5

Table 2 Hyper-Parameters

4. RESULTS AND DISCUSSION

This study uses one of the models from the same Convolutional Neural Network, namely VGG19, with fine tuning and not, which of the two models will be evaluated for its efficiency and accuracy in classifying tomatoes according to their level of ripeness. Based on some of the methodologies discussed above, the VGG19 model training will use the prepared dataset. The comparison was made based on fine-tuning the VGG19 architecture rather than classification accuracy for tomatoes. To maximize accuracy, using hyperparameters such as kernels or the number of filters held by the VGG19 model will impact accuracy and require consideration as an element of performance evaluation for each model. The data set for each model will consist of 300 tomato photos, which will be used throughout the experiment. The dataset is divided into training, validation, and testing data using a 70:15:15 ratio. Of the 300 photos in the dataset, 210 images were chosen for training, 45 for validation, and 45 for testing. The model was tested via iterative training on VGG19 using various kernels. In the next sub-chapters, the performance of several models based on the convolutional neural network technique is compared. The outcomes are displayed in Table 3

Architecture	Batch Size	Kernel	Accuracy
VGG19 Fine Tune	32	3x3	84.44%
	64	3x3	77.78%
VGG19	32	3x3	91.11%
	64	3x3	88.89%
	32	5x5	86.67%
	64	5x5	82.22%

Table 3 Results Of Testing

4.1 Evaluation of VGG19 Fine-Tune

With an accuracy rate of 84.44% and 77.78%, while utilizing Epoch 50 and batch sizes 32 and 64, respectively, VGG 19 produced fairly good results. These accuracy statistics indicate that, in epoch 50, batch size 32 offers a higher level of classification accuracy than size 64. The increased variation in batches with large batch sizes is the cause of batch size's detrimental effect on classification. Additionally, graphs of accuracy and loss resulting from the application of the VGG19 fine-tune 3x3 model are shown in figures 7 and 8.

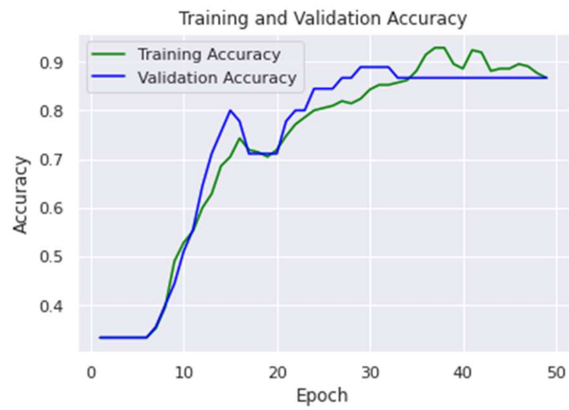

Figure 5 Fine-Tune VGG-19 3x3 Accuracy

Figure 6 Fine-Tune VGG-19 3x3 Loss

4.2 Evaluation of VGG-19

VGG 19 has the highest classification accuracy of 91.11% and 82.22% using epoch 50, kernel 3x3, and batch sizes 32 and 64. In addition, kernel 5x5 produces 86.67% and 82.22% accuracy using parameters epoch 50, batch sizes 32 and 64. With these results, this study obtained data that adding kernel and batch sizes can reduce the accuracy of the VGG 19 model. Besides, a smaller batch size can generalize better than a large batch size, and a 3x3 kernel size gets more optimal results than 5x5. The performance graph of the VGG 19 model with **91.11%** using 3x3 parameters, batch size 32 and epoch 50, is shown in Figure 9, and there is the smallest amount of loss among the other models, which is equal to 0.16, which is shown in Figure 10.

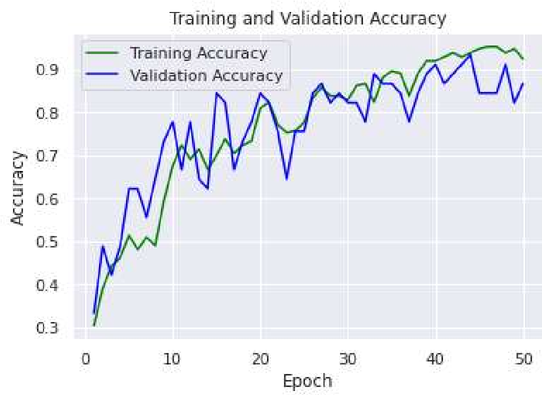

Figure 7 Training and Validation Accuracy Graphic

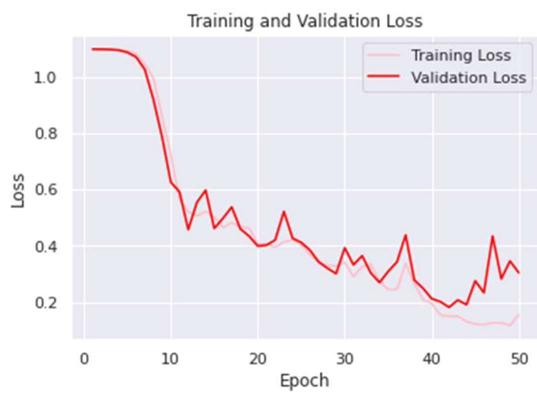

Figure 8 Training and Validation Loss Graphic

5. CONCLUSION

In this study, researchers applied Convolutional Neural Networks to classify tomato images. The CNN models used during pre-training are VGG19 and fine-tuning VGG19, and VGG19 consists of three main layers, namely a convolution layer that will use 3x3 and 5x5 kernels for comparison, the second layer is a max pooling layer, and the last layer is fully connected. In VGG19, fine-tuning will only use the last few layers, as shown in Figure 5, and VGG19 will use all layers from the start node to the end. This study also compares the use of parameters such as batch size and kernel size to get the best model performance. Based on the experiments conducted, larger batch and kernel sizes will decrease the validation value of the model. The classification results of training data on the VGG19 model using a batch size of 32 and a 3x3 kernel get the best result of 91.11% and a classification result of 82.22% using a batch size of 64 and a 5x5 kernel.

So, based on the data from this study, CNN can be used as one of the techniques that can be used to identify the level of maturity, whereas in this study using VGG19 as a model to select the maturity level of tomatoes, and with the presence of automatic image processing can help reduce human errors that can occur when identifying tomato quality. And the use of parameters can also be an important factor in maximizing the performance of the CNN model used.

REFERENCES

[1] AIisyah Hidayattullah, “Identifikasi Tingkat Kematangan BuahTomat(Lycopersicon esculentum Mill) Menggunakan Metode Pengolahan Citra dan Jaringan Saraf Tiruan,” Universitas Gadjah Mada, Yogyakarta, 2013.

[2] S. Nurhayati, “Produksi TanamanTomat(Lycopersicum esculentum Mill.) F1 Hasil Induksi Medan Magnet yang DiInfeksi Fusarium oxysporum f.sp.lycopersici,” 2017.

[3] Badan Pusat Statistik, “Produksi Tanaman Sayuran 2021,” 2021.

[4] S. Dey, *Hands-on image processing with Python : expert techniques for advanced image analysis and effective interpretation of image data*.

[5] B. Ramdhani, E. C. Djamal, and R. Ilyas, "Convolutional Neural Networks Models for Facial Expression Recognition," in *2018 International Symposium on Advanced Intelligent Informatics (SAIN)*, 2018, pp. 96–101. doi: 10.1109/SAIN.2018.8673352.

[6] N. Khunafa Qudsi *et al.*, “Identifikasi Citra Tulisan Tangan Digital Menggunakan Convolutional Neural Network (CNN).”

[7] G. Poggi, C. Sansone, L. Verdoliva, and M. Castelluccio, “Land Use Classification in Remote Sensing Images by Convolutional Neural Networks,” 2015. [Online]. Available:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

https://www.researchgate.net/publication/280696016

[8] S. Sharma, S. Sharma, and A. Athaiya, "Activation Functions In Neural Networks," 2020. [Online]. Available: http://www.ijeast.com

[9] H. Gholamalinezhad and H. Khosravi, "Pooling Methods in Deep Neural Networks, a Review."

[10] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for Simplicity: The All Convolutional Net," Dec. 2014, [Online]. Available: http://arxiv.org/abs/1412.6806

[11] Q. Zhang, M. Zhang, T. Chen, Z. Sun, Y. Ma, and B. Yu, "Recent Advances in Convolutional Neural Network Acceleration," Jul. 2018, [Online]. Available: http://arxiv.org/abs/1807.08596

[12] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Sep. 2014, [Online]. Available: http://arxiv.org/abs/1409.1556

[13] N. Begum and M. K. Hazarika, "Maturity detection of tomatoes using transfer learning," *Measurement: Food*, vol. 7, p. 100038, 2022, doi: https://doi.org/10.1016/j.meafoo.2022.100038.

[14] J. L. Rojas-Aranda, J. I. Nunez-Varela, J. C. Cuevas-Tello, and G. Rangel-Ramirez, "Fruit classification for retail stores using deep learning," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2020, vol. 12088 LNCS, pp. 3–13. doi: 10.1007/978-3-030-49076-8_1.

[15] S. K. Behera, A. K. Rath, and P. K. Sethy, "Maturity status classification of papaya fruits based on machine learning and transfer learning approach," *Information Processing in Agriculture*, vol. 8, no. 2, pp. 244–250, Jun. 2021, doi: 10.1016/j.inpa.2020.05.003.

[16] D. M. Asriny, S. Rani, and A. F. Hidayatullah, "Orange Fruit Images Classification using Convolutional Neural Networks," in *IOP Conference Series: Materials Science and Engineering*, May 2020, vol. 803, no. 1. doi: 10.1088/1757-899X/803/1/012020.

[17] T. Purwaningsih, I. A. Anjani, and P. B. Utami, "Convolutional Neural Networks Implementation for Chili Classification," in *2018 International Symposium on Advanced Intelligent Informatics (SAIN)*, 2018, pp. 190–194. doi: 10.1109/SAIN.2018.8673373.

[18] T.-H. Nguyen, T.-N. Nguyen, and B.-V. Ngo, "A VGG-19 Model with Transfer Learning and Image Segmentation for Classification of Tomato Leaf Disease," *AgriEngineering*, vol. 4, no. 4, pp. 871–887, Oct. 2022, doi: 10.3390/agriengineering4040056.

[19] K. Kualitas Buah Salak dengan Transfer Learning Arsitektur VGG and A. Luthfiarta, “VGG16 Transfer Learning Architecture for Salak Fruit Quality Classification,” *Jurnal Informatika dan Teknologi Informasi*, vol. 18, no. 1, pp. 37–48, 2021, doi: 10.31515/telematika.v18i1.4025.

[20] N. Gupta, "A Pre-Trained Vs Fine-Tuning Methodology in Transfer Learning," in *Journal of Physics: Conference Series*, Aug. 2021, vol. 1947, no. 1. doi: 10.1088/1742-6596/1947/1/012028.