



Identification of Pepper Image Using Convolutional Neural Network (CNN) Deep Learning Method

Fauziah

Jurusan Sistem Komputer Fakultas Ilmu Komputer & Teknologi Informasi,
Universitas Gunadarma, Jl. Margonda Raya No. 100, Pondok Cina, Depok 16424, Indonesia

Email: fauziah87@staff.gunadarma.ac.id

ARTICLE INFO

ABSTRACT

Article history:

Received: Des 27, 2021
Revised: Jan 22, 2022
Accepted: Feb 28, 2022

Keywords:

Image,
Convolutional Neural Networks
(CNN),
Deep Learning.

The Convolutional Neural Network (CNN) method is one of the Deep Learning methods that is currently being developed. The Convolutional Neural Network (CNN) method is one of the Deep Learning methods that is currently being developed. The purpose of this study is to identify the pepper image by applying the Convolutional Neural Network (CNN) Deep Learning method. There are several stages in this technique, the first stage is to prepare the required pepper image data set. The next stage is the preprocessing and sorting of pepper images. Then, the formation of models and system training, the last is to do for system testing. This research uses CNN to recognize pepper images and determine the accuracy value. In this study, 19 pepper images were used as testing data from 76 pepper images used in the training dataset. Pepper testing produces an average value of data testing accuracy of 0, 97756064.

Copyright © 2022 Jurnal Mantik.
All rights reserved.

1. Introduction

Indonesia is a country that has abundant natural resources and is known as an agricultural country. One of the many types of Indonesian wealth is the spices which have the best quality and are well known abroad.

With these conditions it is possible for counterfeiting of pepper to occur and in 2016 quoted from electronic media [merdeka.com,2016] that the Directorate of Special Criminal Investigation of the Metro Jaya Police arrested a dealer of fake coriander and pepper, E in Kosambi, Tangerang Regency, Monday (15 /2). The perpetrator mixed the chemicals H₂O₂ (Hydrogen Peroxide) and NaHCO₃ (Sodium Bicarbonate) into the freshly harvested pepper.

To identify counterfeit pepper, it is necessary to detect an object based on an image or image by implementing deep learning methods. The use of deep learning can be based on several factors such as the use of sufficient training data to be able to learn the characteristics of the training data, making parallel development of computers for high enough performance and the design of network structures and training strategies developed significantly. The Convolutional Neural Network (CNN) method is one of the Deep Learning methods that is currently being developed. The network on CNN is made with the assumption that the input used is an image. This network has a special layer called the convolution layer, where in this layer an input image will be processed based on a predetermined filter. From each of these layers will produce a pattern of several parts of the image which will later be easier to classify, thus making the learning function more efficient to implement.

2. Methods

The first stage is data collection and the process can be seen in Figure 1.



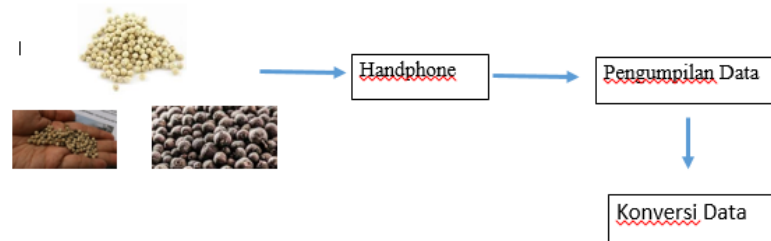


Fig 1. Data collection stage

Figure 1 explains that, in detecting pepper, what must be prepared is the required pepper image dataset. Pepper image data was taken from a smart phone camera as many as 95 photos consisting of 35 white pepper, 50 black pepper and 10 fake peppers. The images were taken in jpg format and collected in a special folder. The data that has been obtained is then converted into pepper image data with varying sizes. All pepper images are used in the training and testing process.

2.1 System planning

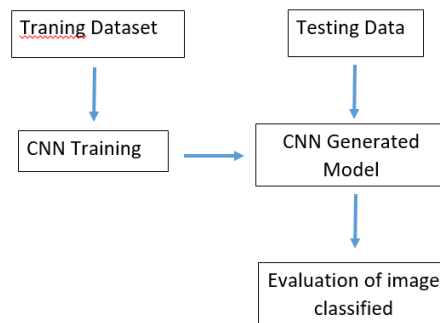


Fig 2. System design

2.2 Training Dataset

The dataset training was taken using own photos from the OPPO F11 type smartphone camera with jpg image format. The total images used are 95 pepper images from 3 classes which are used as training data. The image size will be changed based on the CNN algorithm format to produce a better image. Datasets are labeled according to the file section.

2.3 CNN Training

The labeled image will be processed with the CNN model used is a training with 7 hidden layers (convolutional layer). The workings of the 7 layers are as follows: 6 network layers, with the same architecture each network layer followed by a non-linear ReLU element and a 2 x 2 MaxPooling. learning. In this case study, it takes 5 or 6 iterations (to increase the number of network layers) to find the calculation. The CNN training architecture used in this study can be seen in Figure 3. Stochastic gradient descent was used for CNN training using a small and the same random data set for each iterative learning phase.

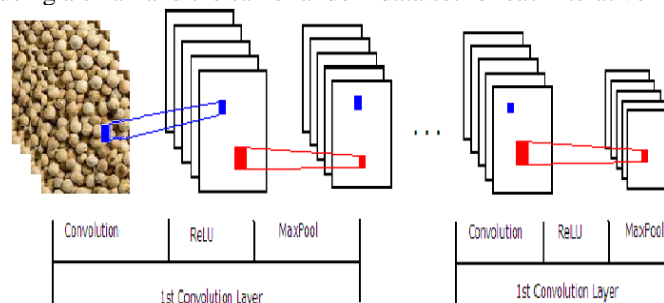


Fig 3. Architecture CNN

2.4 CNN Generated Model

The model generated by CNN is stored to contain the considered data. The CNN model can then be used at any time with experimental data to evaluate the accuracy of the model. The model that will be used is

a simple Multi Layer Perceptron (MLP) model, therefore the Sequential model is used. The CNN model has several layers including convolution, pooling dropout. Can be seen in Figure 4.

Model: "sequential_1"

Layer (type)	Output Shape	Param #
rescaling_2 (Rescaling)	(None, 180, 180, 3)	0
conv2d_3 (Conv2D)	(None, 180, 180, 16)	448
max_pooling2d_3 (MaxPooling2D)	(None, 90, 90, 16)	0
conv2d_4 (Conv2D)	(None, 90, 90, 32)	4640
max_pooling2d_4 (MaxPooling2D)	(None, 45, 45, 32)	0
conv2d_5 (Conv2D)	(None, 45, 45, 64)	18496
max_pooling2d_5 (MaxPooling2D)	(None, 22, 22, 64)	0
flatten_1 (Flatten)	(None, 30976)	0
dense_2 (Dense)	(None, 128)	3965056
dense_3 (Dense)	(None, 3)	387

Total params: 3,989,027
 Trainable params: 3,989,027
 Non-trainable params: 0

Fig 4. Model obtained

In flatten_1 produces an output of 30976 which is obtained from the multiplication calculation of the previous dimensions, namely $22 \times 22 \times 64 = 30,976$ and dense_2 the output is 128 which is a number that shows the number of neurons used. The parameter of 3,965,056 is obtained from the product of $30,976 \times 128 = 3,965,056$. Dense_3 shows the number of image categories used, so that the parameter becomes $3 \times 128 + 3 = 387$. The total parameter of the model is 3,989,027. then conduct data training on the pepper image into a model with a fit model. To fit the model, use epoch = 10, back_size = 32 and validation_split = 0.2. Epoch is the number of times the network will see the entire data set, batch_size is the number of training instances in one forward/backward pass. And it can be concluded that the higher the batch_size value, the more memory is required. So the complete CNN Architecture Model for pepper classification adds a full connection network layer to the training architecture which can be seen in Figure 5. As the final classifier, the Softmax function is used.

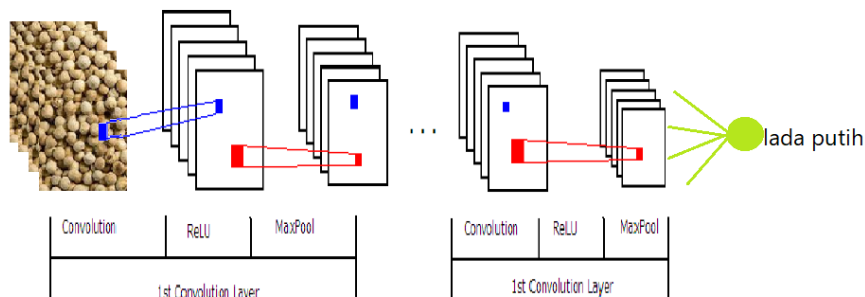


Fig 5. CNN architecture on pepper classification

2.5 Testing Data

In the test using 19 different pepper image data. This study uses the same training setting parameters, namely the maximum epoch value is 10. Convolution2dLayer parameter settings use the image length and width of filter 2 and filter number 10 and maxPooling2dLayer with pool size 2 and Stride 2. Experimental data is a set of pepper images. This data is input for the model obtained from CNN to predict

the level of accuracy in the image which may be a similar image of white pepper, black pepper and fake pepper.

2.6 Evaluation Of Image Classified

At this stage is the stage that evaluates the image that has been classified according to the results of the level of accuracy and suitability of the pepper image. To compare the value of accuracy can be done with the concentration of the addition of the number of layers on the CNN architecture. And this addition will not change the value of predefined parameters such as epoch value and learning rate. The convolutional layer used is layer seven, the accuracy value is obtained after the seventh layer which is the average of each layer having an accuracy value. In this study, experimental data are input for the resulting model and can identify the image of white pepper, black pepper and fake pepper which is the result of the output. The application used in this research uses the framework from TensorFlow. TensorBoard is a histogram from TensorFlow, a graphical visualization tool that makes it easy to understand model parameters and their variations over time.

3. Results and Analysis

3.1 Data Collection Results

In taking the following data are the results of training obtained from the CNN model and can be seen in the graphs of Figures 11 and 12.



Fig 6. Grafik Training dan Validation Accuracy

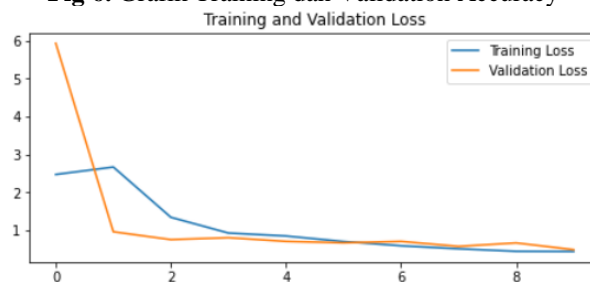




Fig 7. Grafik Training dan Validation Loss

From the results of the process carried out, namely the training and testing process through the CNN method using the validation of the training and testing process. And the test results can be seen in table 1 for white pepper, table 2 for black pepper and table 3 for fake pepper.

Table 1.
White Pepper Test Results

Citra	Ukuran	Akurasi
 LadaPutih (2).jpg	225x225	83.45 %
 LadaPutih (12).jpg	275x183	84.11%

Citra	Ukuran	Akurasi
 LadaPutih (14).jpg	258x195	75.16%
 LadaPutih (33).jpg	224x224	56.98%
 LadaPutih (35).jpg	700x395	88.73%

Table 2.
Black Pepper Test Results

Citra	Ukuran	Akurasi
 ladahitam (1).JPG	336x168	99.92%
 ladahitam (5).JPG	188x172	99.85%
 ladahitam (14).JPG	514x421	99.91%
 ladahitam (22).JPG	210x201	97.92%
 ladahitam (48).JPG	304x181	96.86%

Table 3.
Fake Pepper Test Results

Citra	Ukuran	Akurasi
 ladapalsu 2.JPG	210x169	47.6 %
 ladapalsu 3.JPG	149x99	46.06%

Citra	Ukuran	Akurasi
 ladapalsu 6.JPG	389x372	51.24%
 ladapalsu 7.JPG	379x212	43.46%
 ladapalsu 9.JPG	521x282	37.7%

From the results of observations, the 3 data that have been taken illustrate that the accuracy that is close to 100% is black pepper where from its size black pepper itself has a different size from white pepper and fake pepper. And the accuracy value below 50% is fake pepper with different sizes from white pepper and black pepper. This proves that the system created is capable of detecting fake peppercorns and genuine peppers from smartphone camera images.

So from the overall data it can be concluded that the accuracy value obtained from training and testing. The data obtained is 99% accuracy value from testing 76 pepper images and testing data has an accuracy value of 97% with 19 images from pepper images. The accuracy value of the training and testing process using the CNN method from a predetermined image can be seen in table 4.

Table 4.

Results of Classification Accuracy of Training and Testing

Data	Jumlah Data	Akurasi (%)
Traning	76	99
Testing	19	97

4. Conclusions

Based on the results of the analysis that has been carried out, several conclusions are obtained, namely the level of accuracy of the testing data obtained from the model formed is 99% in classifying pepper images. The results of the classification of the new data as many as 5 pepper images to test the formed model, namely all the images were identified correctly. Suggestions in efforts to improve and develop, then the next research can add image data that is used to train the model and achieve a high level of accuracy. Using the latest image classification method with better accuracy.

References

- [1] D. Wang et al., "Modulation Format Recognition and OSNR Estimation Using CNN-Based Deep Learning," in *IEEE Photonics Technology Letters*, vol. 29, no. 19, pp. 1667-1670, 1 Oct.1, 2017, doi: 10.1109/LPT.2017.2742553.
- [2] F. Chen and M. R. Jahanshahi, "NB-CNN: Deep Learning-Based Crack Detection Using Convolutional Neural Network and Naïve Bayes Data Fusion," in *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4392-4400, May 2018, doi: 10.1109/TIE.2017.2764844.
- [3] M. Duan et al., "A Hybrid Deep Learning CNN-ELM for Age and Gender Classification," in *Neurocomputing*, vol. 275, pp. 448-461, 1 Jan.31, 2018, doi: 10.1016/j.neucom.2017.08.062.
- [4] X. Yang, Y. Ye, X. Li, R. Y. K. Lau, X. Zhang and X. Huang, "Hyperspectral Image Classification With Deep Learning Models," in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 9, pp. 5408-5423, Sept. 2018, doi: 10.1109/TGRS.2018.2815613.
- [5] F. Milletari et al., " Hough-CNN: Deep learning for segmentation of deep brain regions in MRI and ultrasound," in *Computer Vision and Image Understanding*, vol. 164, pp. 92-102, 1 Nov, 2017, doi:

10.1016/j.cviu.2017.04.002.

- [6] Isna Wulandari, Hasbi Yasin, Tatik Widiharih, 2020, Klasifikasi Citra Digital Bumbu dan Rempah Dengan Algoritma Convolutional Neural Network (CNN), *Jurnal Gaussian*, Vol 9, No. 3, pp. 273 – 282.
- [7] Zhao, Z.-Q., Zheng, P., Xu, S., & Wu, X. 2018. Object Detection with Deep Learning: A Review.

