Deep Learning Implementation using Convolutional Neural Network in Mangosteen Surface Defect Detection

by Laila Azizah

2017 7th IEEE International Conference on Control System, Computing and Engineering (ICCSCE 2017), 24–26 November 2017, Penang, Malaysia

Deep Learning Implementation using Convolutional Neural Network in Mangosteen Surface Defect Detection

Laila Ma'rifatul Azizah
Information Technology
Universitas Muhammadiyah
Yogyakarta (UMY)
Yogyakarta, Indonesia
laila.m.azizah@umy.ac.id

Sitti Fadillah Umayah
Information Technology
Universitas Muhammadiyah
Yogyakarta (UMY)
Yogyakarta, Indonesia
sitti.fadillah.2013@ft.umy.ac.id

Slamet Riyadi
Information Technology
Universitas Muhammadiyah
Yogyakarta (UMY)
Yogyakarta, Indonesia
riyadi@umy.ac.id

Cahya Damarjati
Information Technology
Universitas Muhammadiyah
Yogyakarta (UMY)
Yogyakarta, Indonesia
cahya.damarjati@umy.ac.id

Nafi Ananda Utama
Agrotechnology,
Universitas Muhammadiyah
Yogyakarta (UMY)
Yogyakarta, Indonesia
nafi@umy.ac.id

***Abstract*— Mangosteen is one of the fruits that has an enormous export potential in Indonesia. However, not all mangosteen is the defect free fruit. The quality assurance in mangosteen export is done manually by sorting expert. Therefore, this can lead unstandardized and inaccurate results. The result happens because of human error. It needs an image processing technology to help the sorting process which one is the defect and non-defect. In this study, we use one of deep learning architecture that is Convolutional Neural Network (CNN). Therefore, we use CCN as a detection of mangosteen. CNN proved to be very efficient regarding classifying images. This CNN method is implemented using 4-folds Validation Cross to validate data accuracy. In the preparation of the CNN architecture model, initializing the parameter configuration accelerates the network training process. The results of the experiments using CNN algorithm showed the performance of defect detection on the mangosteen fruit of 97%.**

***Keywords*— deep learning; convolutional neural network; image processing; mangosteen; surface defect detection.**

I. Introduction

Quality control of mangosteen is the measuring value of mangosteen whether it is feasible or not to be marketed. Workers still do quality inspection conventionally who are utilizing the sense of sight and touch. The conventional way is a less efficient way because it uses a lot of time and money to hire workers. Therefore, it takes a system that can check the quality of the mangosteen surface.

The system can identify the defect or absence of the surface on the mangosteen directly from the surface image of the mangosteen itself. Firstly, figure or image of mangosteen surface skin is processed by computer with image processing. Image processing is a process that change and represent real image to a matrix or number value. Thus image is manipulated and calculated the value to get recognition that user need[1].

Suchitra A. Khoje developed a evaluation methodology for quality of fruit surfaces with the classification of Probabilistic Neural Network (PNN) and Support Vector Machine (SVM) [2]. PNN is one of artificial neural network architecture.

Deep learning is subset of machine learning that using the artificial neural network for pattern recognition[3]. In deep learning, a computer learns to classify directly from images, text, or sound. The computer is trained to use big data sets and then change the pixel value of the picture to an internal representation where the classifier can detect patterns on the input [4].

One of the most frequently used deep learning algorithms concerning the image classification is the convolutional neural network (CNN). CNN is used to classify the data that are labeled using supervised learning method, where the workings of supervised learning are data trained, and targeted variables so that the purpose of this approach is to group data into existing data[4,5].

CNN is often used to recognize objects or landscapes and perform object detection and segmentation. CNN learns directly from the image data, thus eliminating manual feature extraction. The first research was undertaken by Hubel and Wiesel who conducted visual cortex research on the cat's vision of vision. Visual cortex in animals is very powerful in visual processing systems that ever existed. The animals' visual cortex inspires many studies. Some researchers produce new models such as Neocognitron, HMAX, LeNet-5, and AlexNet.

The application of the CNN method was carried out in R. Socher et al. In his research, "Convolutional-Recursive Deep Learning for 3D Object Classification" integrates convolutional and recursive neural networks as a learning feature and RGB-D image classification [6]. Other studies have also been conducted by I Wayan Suartika E. P. et al. who applied CNN to classify object images [7]. M.Zufar also applied CNN for Face recognition[8].

The deep learning method with the convolutional neural network architecture is used in this study to detect any defect on the surface of the mangosteen. We hope that this research can help to improve the quality and export levels. The detection of the mangosteen defect surface could impact the reducing in time and the amount of expenditure on the export of mangosteen.

II. Methodology

This research uses Convolutional Neural Network (CNN) as our method. First, Expert sort out the mangosteen manually. This step is used as primary data to define wheater the mangosteen is in good shape or not. We labeled the mangosteen data as the image "fine" and image "defect". Then, the labeled mangosteen image is cropped and resized before the classification process. We resize and crop the mangosteen image in 512 x 512 square picture which only shows full mangosteen surface. The next step is the classification process that is done using CNN architecture. This stage uses training phase first then the testing phase. The training phase is to train which image is labeled as "defect" and " fine". On the other hand, the testing process is the primary process to classified the mangosteen image as "defect" or "fine" one from training data.

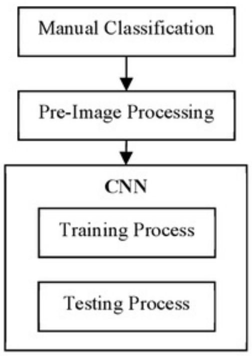

Fig. 1. The Classification Process

A. Data Acquisition

Primary data in this data acquisition stage is the mangosteen image was taken. The mangosteen itself have been classified manually and taken as samples. The images are sorted according to the research need as data. After the image is sorted, the sample size is tailored to the needs. Besides, the image data is also divided into two groups as data training and data testing.

Image data is captured using a digital camera with the same distance, illumination and image resolution of 6000 x 4000 pixels. On the data retrieval, in one photo contains two mangosteens.

B. Pre-Image Processing

The aim of this stage is to make uniform image data input before the image is being processed using CNN (convolutional neural network) classification method. However, in the image taking process, there is a possibility where the size of real mangosteen and the distance of image taking would affect the result. Therefore, there is a need for cropping and resize the image. Because the CNN classification method requires uniform size in the input image, the resize function is used to make consistent image size. We crop and resize the image to 512x512 pixels. Thus resize function would help computation faster in CNN. Figure 2 is the process of cropping and resizing the mangosteen image.

Fig. 2 Pre-Image Processing before Classification Process

Figure 3 shows image data labeled 'fine' that has been through the pre-image processing process. This figure is the figure of good quality mangosteen skin surface. Figure 4 shows image data marked 'defect' that has been through the image pre-processing process. Figure 4 is the figure of mangosteen skin surface that has wound in the skin.

Fig. 3 Image data with 'fine' category

Fig. 4 Image data with 'defect' category

C. Training

The training stage is the stage where CNN is trained to get high accuracy when performing classification. The training process is shown in figure 8. There feedforward process and backpropagation process in this stage. The feedforward process requires the number and size of the layer that is needed, the size of subsampling, the parameters of the training settings, as well as the image data obtained. The parameters determine how many programs train the computer to study the data as well as how many layers will be used. Therefore, the parameters can identify the accuracy of the program in the classification process. The number of epoch determines how the feedforward and backpropagation process will be performed in a single training; the mini batch determines how many iterations of image data will be handled in 1 epoch. Iteration (i) is determined by:

$$i = \frac{\sum training\ data}{batch} \times \sum epochs \qquad (1)$$

In this case, we use the primary layers which are:

1) *Convolution Layer* does a convolution stage on the previous layer output. This layer is the primary stage that underlies CNN. Convolution is a mathematical term that means applying a function to the output of another function repeatedly. In image processing, convolution means to implement a kernel (yellow box) to the image in all possible offsets which shown in figure 5.

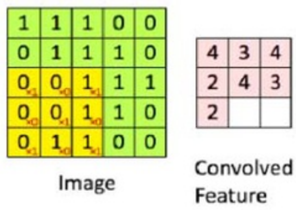

Fig. 5 Operation on convolution[9]

2) *Subsampling Layer* performs the process of reducing the size of an image data. In image processing, subsampling also aims to increase the position invariance of features. In most CNN, the subsampling method uses max pooling. Max pooling divides the convolution layer output data into several small grids and then gets the maximum number of each grid to construct a reduced image matrix. The use of a pooling layer on CNN only aims to minimize the size of the image. Therefore it can be easily changed by a convolution layer with the same stride value as the corresponding pooling layer[10]. Figure 6 below is operation on the max pooling process to get a reduced matrix.

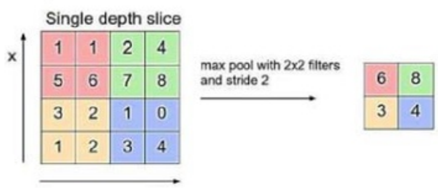

Fig. 6 Operation on max pooling[11]

3) *The fully connected layer* is a layer commonly used in MLP implementation and aims to transform the data dimension so that data can be classified linearly. Each neuron of the convolution layer requires transformation process into one-dimensional data set first before it can be inserted into a fully connected layer. The fully connected layer can only be implemented at the end of the network because this process causes spatial data loss and the process is not reversible.

4) *The softmax layer* is an activation layer. The softmax layer is used when the problem multiclass classification, the output layer usually has more than one neuron. Softmax layers are normalized exponential probabilities of class observations represented as neuron activation. The exponential function increases the probabilities of the maximum value of the previous layer compared to the other values.

The feedforward step is the first step in the training stage. The feedforward step pulls the vector through the convolution process and max pooling. This work is to reduce the size of its image and multiply its neurons. Many networks are formed, which adds variants of the data to be studied. Figure 7 is feedforward step which is concluded that the process will produce multiple layers to classify the data using updated biases and weights from the backpropagation stage.

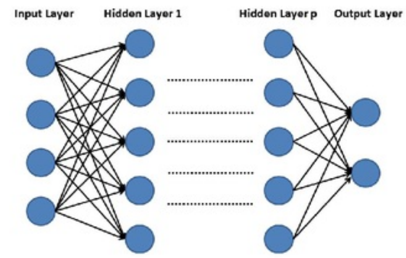

Fig.7. Feedforward process with layers by layers

Backpropagation step is the second step of the training stage. At this process, the feedforward results trace error from the output layer backward to the input layer. To indicate that the data has been traced, new weights and bias are obtained.

The gradient calculation network for convolutions produces weight and bias number. That number will be used in the training process. Gradient descent algorithm updates the parameters to lessen the error by taking small action in the direction of the loss function in negative gradient, $\nabla E(\theta)$ [12].

$$\theta_{\ell+1}=\theta_{\ell}-\alpha\nabla E(\theta_{\ell}) \qquad (2)$$

Where ℓ stands for the number of iteration, $\alpha>0$ is the learning rate, θ is the vector of the parameter, and $E(\theta)$ is the loss function.[13] The gradient of the loss function, $\nabla E(\theta)$, is evaluated by employing all training set, and the standard gradient descent algorithm utilizes all data set. The algorithm of stochastic gradient descent puts the gradient, therefore updates the parameters, taking a subset of the training data. This subset is called a mini batch.

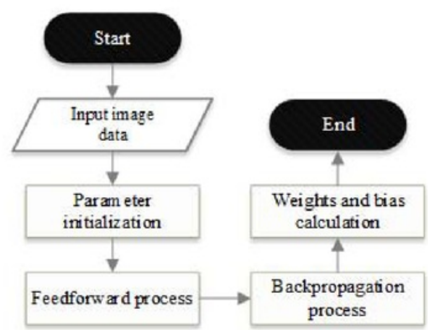

Fig. 8. Training process

D. Testing

The testing process is a classification process using the bias and weight from the results of the training stage. The testing process is dissimilar from the training process which the big distinguishes is no back propagation step after the feedforward step. Therefore, at the end of this stage results; the performance classification accuracy, the failed classification data, the image number of failed classification, and the model of a network from the feedforward stage. The feedforward process generates an output layer with new weights and biases. The output layer is connected with the provided label. These fully connected results are obtained from success data rate that had been classified.

E. Validation

The validation process of this research is using K-fold Cross Validation. Cross validation calculates and clarifies the data that haven't in the data set. In the model data set, the training data as primary data and validate with testing data. Figure 9 is the example of 4-fold Cross Validation. In this method, the data is divided into four fold and processed one by one per fold. The accuracy mean of all folds generates the overall accuracy value.

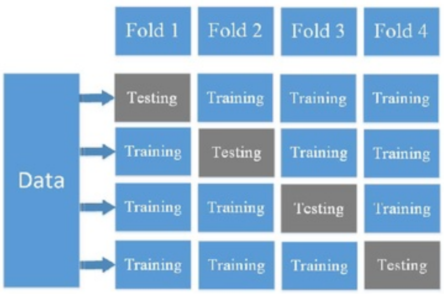

Fig. 9. 4-fold Cross Validation

III. Result

This research conducts with 120 test images (30 defect images and 90 fine images) which are divided into 4-fold data sets using deep learning classification with CNN algorithm and 4-Fold Cross Validation as image classifiers. Each of the folds is tested using a designed detection program. One of the image data examples of the fold is in figure 10. The folds generate an overall percentage accuracy that can be an indicator of the success rate.

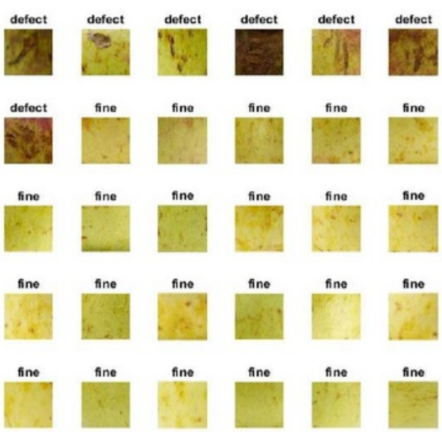

Fig. 10. Testing results with error free for fold 1 data set

The previous training stage influences the classification results. In the testing phase, fold one can process the classification well until it reaches 100%. However, the accuracy of the training process tends to be less stable. Figure 11 shows the table from the result of the training process fold one.

Epoch	Iteration	Time Elapsed (seconds)	Mini-batch Loss	Mini-batch Accuracy	Base Learning Rate
3	50	7.54	3.1885	80.00%	0.000100
6	100	15.09	-0.0000	100.00%	0.000100
9	150	22.70	-0.0000	100.00%	0.000100
12	200	30.28	-0.0000	100.00%	0.000100
14	250	37.83	3.1885	80.00%	0.000100
17	300	45.37	3.1885	80.00%	0.000100
20	350	52.95	-0.0000	100.00%	0.000100
23	400	60.52	-0.0000	100.00%	0.000100
25	450	68.10	-0.0000	100.00%	0.000100
28	500	75.68	3.1885	80.00%	0.000100
31	550	83.25	-0.0000	100.00%	0.000100
34	600	90.79	-0.0000	100.00%	0.000100
37	650	98.35	-0.0000	100.00%	0.000100
39	700	105.96	3.1885	80.00%	0.000100
42	750	113.50	3.1885	80.00%	0.000100
45	800	121.06	-0.0000	100.00%	0.000100
48	850	128.60	-0.0000	100.00%	0.000100
50	900	136.16	-0.0000	100.00%	0.000100

Fig. 11. Training results for fold one data set

On the training process, the number of layers consists of the input layer as the first input, the convolution layer for the next layer, subsampling layer, fully connected layer, and the last is softmax layer as activation, and classification output layer. We implement the same parameters as we did in the training process. The maximum number of the epoch is 50, and the size of the mini batch is 5. The performance of CNN can get the highest with filter width 2 and filter number 10 in layer setting. The pool size is 2, and the stride of it is 2.

This parameter is determined by testing some values that can be input for each parameter. In the convolution layer and max pooling layer, we performed some trial testing phases to determine which size can produce optimal accuracy. The convolution layer parameters tested with the filter width 2 and filter number 10 then the filter width size 5, and filter number 20. While the max pooling layer parameters tested in the size of pool 2 and Stride 2 then the size of pool 4 and Stride 2. Table 1 shows the result of layer testing. We found that the convolution layer with filter width 2 and filter number 10 then on max pooling layers with pool size 2 and Stride 2 are the correct parameters to produce high accuracy.

TABLE I. Result of Layers testing Parameter

Conv \ Pool	(2, S2)	(4, P2)
(2,10)	90%	80%
(5,20)	86.67%	70%

The next step is choosing the right epoch and mini batch. We tried with 15, 30, and 50 epoch and 2, 5, 10 mini batch. As shown in table 2, the best combination from our trial is 50 epoch and 5 mini-batch.

TABLE II. Result of Layers Parameter Testing

Epoch \ Batch	2	5	10
15	90%	86.67%	76.67%
30	70%	70%	60%
50	8.67%	96.67%	70%

Table 3 is the classification results of 120 testing images that divided by 4-fold classification. This algorithm can get 97.5% of the average result in detecting mangosteen defect surface.

TABLE III. THE RESULT OF CLASSIFICATION WITH PERCENTAGE ACCURACY

Algorithm	Percentage Accuracy %				**Mean**
	fold-1	*fold*-2	*fold*-3	*fold*-4	
CNN	100%	100%	93.33%	96.67%	97.5%

IV. CONCLUSIONS AND FUTURE WORK

Our proposed method to detect the mangosteen defect surface can reach 97.5% optimal accuracy. Although there is an error reading in the system, it is not more than 2.5% in average. Error reading occurs due to the lighting intensity factor at the time of data retrieval. This factor affects the result of image extraction.

Future research is expected to test with other parameters and combination of feature extraction of full picture of mangosteen skin surface on CNN classification to get the most suitable for the classification process. Therefore, this can produce optimum accuracy.

V. ACKNOWLEDGEMENT

Authors would like to acknowledge Ministry of Research, Technology and Higher Education, Republic of Indonesia for the grant Penelitian Unggulan Perguruan Tinggi 2017.

REFERENCES

[1] R.C. Gonzalez, and R.E. Woods, "Digital Image Processing", New Jersey: Pearson Education, 2002.

[2] S.A. Khoje, S.K. Bodhe, and A. Adsul, "Automated Skin Defect Identification System for Fruit Grading Based on Discrete Curvelet Transform" , International Journal of Engineering and Technology (IJET), 5(4), 3251-3256, 2013.

[3] K.N. Ranjit, H.K. Chethan, and C. Naveena, "Identification and Classification of Fruit Diseases", International Journal of Engineering Research and Application (IJERA), pp.11 – 14), 2016.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning" Nature, 521 (7533), pp. 436-444, 2015.

[5] Mathworks. "Convolutional Neural Network". *Mathworks.* https://www.mathworks.com/discovery/convolutional-neural-network.html.

[6] R. Socher, B. Huval, B. Bath, C.D. Manning, and A.Y. Ng, "Convolutional-Recursive Deep Learning for 3D Object Classification" , In Advances in Neural Information Processing Systems (pp. 656-664), 2016.

[7] W.S.E. Putra, "Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101", Jurnal Teknik ITS, *5*(1), 2016.

[8] M. Zufar, and B. Setiyono, "Convolutional Neural Networks untuk Pengenalan Wajah Secara Real-Time", Jurnal Sains dan Seni ITS, *5*(2), 2016

[9] http://deeplearning.stanford.edu/wiki/index.php/Feature_extraction_using_convolution

[10] J.T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for Simplicity: The All Convolutional Net", ICLR-2015, 2014.

[11] C.M Bishop, "Pattern Recognition, and Machine Learning", Springer, New York, NY, 2006.

[12] http://cs231n.github.io/convolutional-networks/

[13] https://www.mathworks.com/help/nnet/ref/trainingoptionssgdm-class.html#bvbjl7f-6

Deep Learning Implementation using Convolutional Neural Network in Mangosteen Surface Defect Detection

ORIGINALITY REPORT

13%	7%	11%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	la.mathworks.com Internet Source	3%
2	roar.uel.ac.uk Internet Source	2%
3	Renaldi Primaswara Prasetya, Fitra A. Bachtiar. "Indonesian food items labeling for tourism information using Convolution Neural Network", 2017 International Conference on Sustainable Information Engineering and Technology (SIET), 2017 Publication	2%
4	Agus Khumaidi, Eko Mulyanto Yuniarno, Mauridhi Hery Purnomo. "Welding defect classification based on convolution neural network (CNN) and Gaussian kernel", 2017 International Seminar on Intelligent Technology and Its Applications (ISITIA), 2017 Publication	2%
5	Sulis Setiowati, Zulfanahri, Eka Legya Franita,	

Igi Ardiyanto. "A review of optimization method in face recognition: Comparison deep learning and non-deep learning methods", 2017 9th International Conference on Information Technology and Electrical Engineering (ICITEE), 2017
Publication

1%

---

6 A. Sivaranjani, S. Senthilrani, B. Ashokumar, A. Senthil Murugan. "An Improvised Algorithm For Computer Vision Based Cashew Grading System Using Deep CNN", 2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA), 2018
Publication

1%

---

7 Tony K. Hariadi, Hasan Zidni, Karisma Trinanda Putra, Slamet Riyadi. "High Accuracy Real Time Machine Vision for Diameter Measurement Using Simpson Algorithm", 2018 International Conference on Information and Communication Technology Convergence (ICTC), 2018
Publication

1%

---

8 macrothink.org
Internet Source

<1%

---

9 docplayer.net
Internet Source

<1%

10	ijies.net Internet Source	<1%
11	www.irjet.net Internet Source	<1%
12	cs.stanford.edu Internet Source	<1%
13	Wang, Xiaolong, and Chandra Kambhamettu. "Age estimation via unsupervised neural networks", 2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), 2015. Publication	<1%
14	"Artificial Neural Networks and Machine Learning – ICANN 2018", Springer Nature America, Inc, 2018 Publication	<1%
15	Gao, Xinting, Stephen Lin, and Tien Yin Wong. "Automatic Feature Learning to Grade Nuclear Cataracts Based on Deep Learning", IEEE Transactions on Biomedical Engineering, 2015. Publication	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		