International Conference On Agriculture and Applied Science (ICoAAS) 2022 **EISSN : 2776-043X**
Politeknik Negeri Lampung, 15 November 2022 Doi: https://doi.org/10.25181/icoaas.v3i3.2867

Potato Leaf Disease Detection with Convolutional Neural Network Method

TS Jaya [1] and D Sahlinal [1]

[1]Informatics Management Study Program, Politeknik Negeri Lampung, Indonesia

*e-mail: sandi@polinela.ac.id

Abstract. Potatoes are recognized by people as a staple food abroad because they contain carbohydrates. In Indonesia, potatoes are still considered a luxury vegetable. However, potatoes are a portion of delicious and very nutritious food. One of the main diseases that attack potato plants is late blight and another disease that is often encountered in potato plants is dry blight (early blight). In this research, we develop software to detect potato leaf disease using the convolutional neural network method. The advantage of the convolutional neural network method is that it can automatically extract important features from each image without human assistance, besides that the convolutional neural network method is also more efficient than other neural network methods, especially for memory and complexity. This study resulted in an application with all functions running correctly based on the results of the BlackBox test.

1. Introduction

Potato plants grow well in highland or mountainous areas with a slope of 800-1,500 meters above sea level (asl). Potato plants are one of the plants that are susceptible to disease and need to be treated by social groups, researchers, or related organizations. Problems arise in the community with a lack of attention from stakeholders, so sometimes farmers' harvests are not satisfactory and lead to a decrease in harvest quality and quantity [1].

The problem of potato leaf disease has been addressed not only in agriculture but also in technology. One of them is by utilizing informatics to identify diseases that occur in potato plants using image processing or processing commonly referred to as processing. , digital images. Using image processing for identification enables farm managers to effectively and efficiently manage unhealthy or abnormal crops. With the development of today's technology, there have been many studies that have developed digital imaging in agriculture to identify diseases from agricultural products [2]. One way is to identify potato leaf disease. The most common potato disease is *late blight*, but there is also *early blight*. Cold and humid places are one of the factors that cause disease outbreaks.

This study aims to create a system that can help farmers and farmers to identify potato leaf disease using potato leaf image data. Identification of potato leaves is divided into three parts: potato plants with healthy or normal leaves, late blight, and early late blight. Therefore, in this study, this classification was carried out using a web-based application by applying the convolutional neural network (CNN) algorithm.

2. Basic theory

The method using the Convolutional Neural Network (CNN) is one of the methods of machine learning which is the development of the Multi-Layer Perceptron (MLP) which is designed to process or create data from two dimensions [3]. CNN is also one type of method from the Deep Neural Network because it has a network level and has many applications carried out in the image. The CNN method consists of two methods, namely classification using feedforward and the learning stage using backpropagation [4]. The working principle of this method is similar to the MLP method, but in the CNN method, each neuron is presented in two dimensions which is not the same as the MLP method where each neuron only has one dimension.

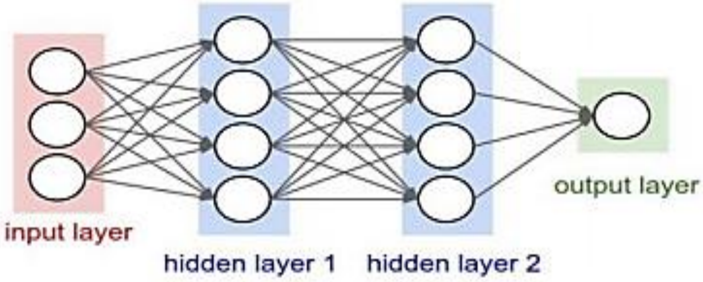

Figure 1. Multilayer Perceptron Architecture

The MLP method which can be seen in the image above it has I layers with each layer having Ji neuro an. The MLP method accepts input data in one dimension and propagates the data to the network to produce the output [5]. Each connection between neurons in two neighboring layers has a one-dimensional weight value that determines the quality of the mode. Each input data layer is calculated with the existing weight values, then the calculation results will be transformed using a non-linear calculation which is called the activation function. The data propagated into the CNN method is data with two dimensions, so the calculations are carried out linearly and using different weight parameters on CNN. Linear calculations in the CNN method use convolution calculations, with weights that are no longer just one dimension, but are already in the form of four dimensions which are a collection of various convolution kernels as shown below. The dimensions of the weights in the CNN method are:

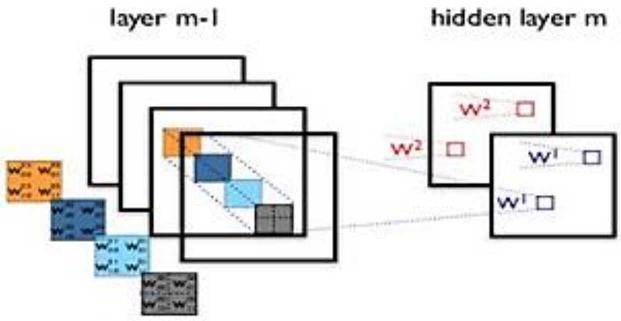

Figure 2. Convolutional Neural Network Method Process

In the CNN method, the input layer used previously is not a one-dimensional form but a two-dimensional form [6]. If it is analogous to the features of the human face, the first layer is the depiction of strokes in different directions whereas in the second layer the features such as the shape of the eyes, nose, and mouth begin to appear, the process is due to the merging of the first layer which is still in the form of scratches. In the third layer, a combination of the features of the eyes, nose, and mouth will be formed which will later be concluded with the face of a certain person and the results may even be identified. The CNN method has several hidden layers from an input that has a single vector [7]. Inside the input is a digital image that is converted into a single vector. Inside the hidden layer, several neurons seem to have four mapping features, namely C11 in the image. In C1 neurons are connected to neurons in S1, and so on. The last layer that is connected to the previously hidden layer is called the output layer and the final result is presented in class classification [8].

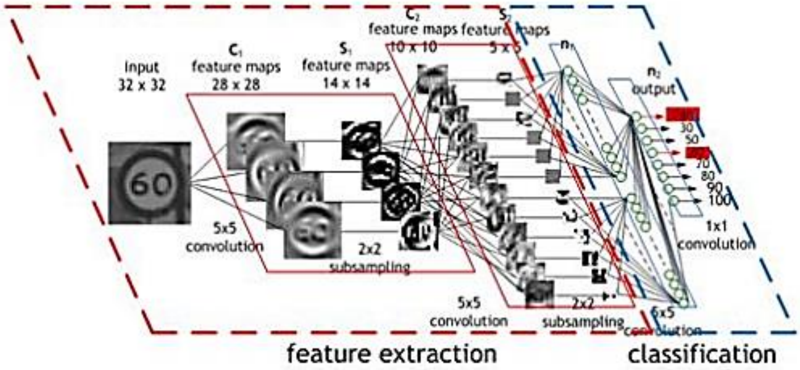

Figure 3. Convolutional Neural Network Architecture

3. Methods

This study uses the method shown in Figure 4.

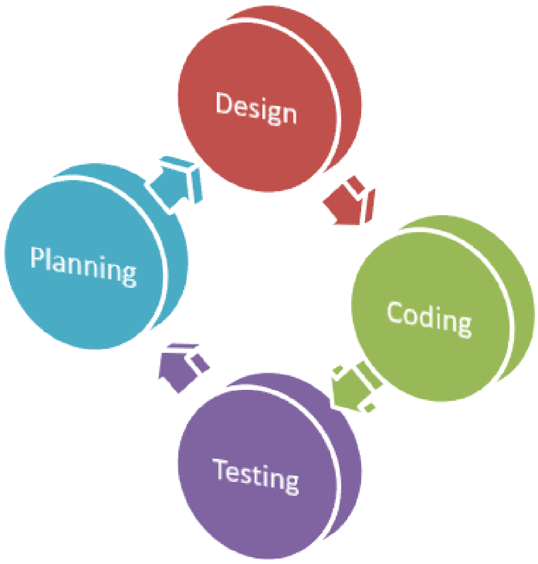

Figure 4. Research Method

The explanation of each stage is as follows.

3.1. Planning

In the planning process, the sample data used was collected. The data used uses public datasets available on the Kaggle.com web.

3.2. Design

In this process, the system design is carried out with the unified modeling language (UML) modeling language and user interface design.

3.3. coding

In this process, coding is carried out using PHP and the python library.

3.4. Testing

In this process, the application is tested using the black box method. In this process, scenarios that represent the function of the application will be tested.

4. Results and Discussion

4.1. Planning

The dataset used was taken from the Kaggle.com website. The dataset is divided into three classes, namely healthy leaves, leaves with early blight disease, and leaves with late blight disease. An example of a healthy potato leaf image is shown in the figure.

Figure 5. Healthy Potato

An example of an image of a potato leaf with early blight is shown in the figure.

Figure 6. Potatoes with Early Blight

An example of an image of a potato leaf with late blight is shown in the figure.

Figure 7. Potatoes with Late Blight

In the dataset used, the data is divided into 2, namely training data and test data. The separation of data groups used in the training process is 80% and the data in the test or testing process is 20% of the total data. The training data stage is where CNN is trained to obtain higher accuracy.

4.2. *Design*

This process produces process modeling with a unified modeling language (UML) and application interface design.

4.2.1 *Process Modeling.*

Figure 8 shows the Use Case Diagram of the application. The use case shows that there are 3 major processes, namely viewing information, making predictions, and seeing the results of recommendations.

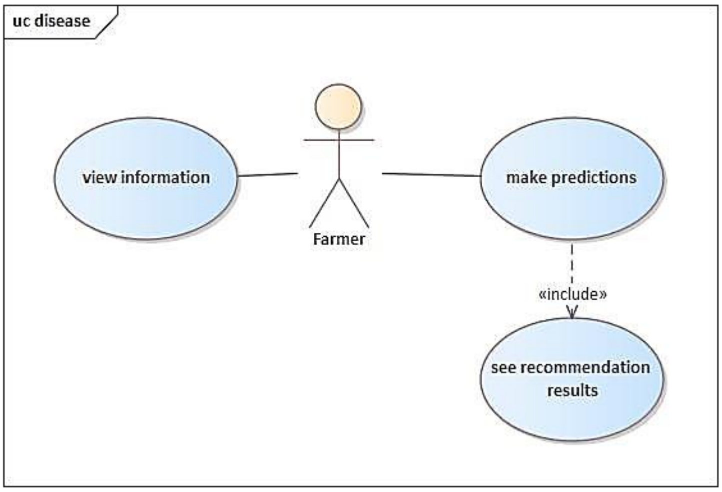

Figure 8. Use Case Diagram

4.2.2 *Interface Design*

This stage produces the interface design used in the application.

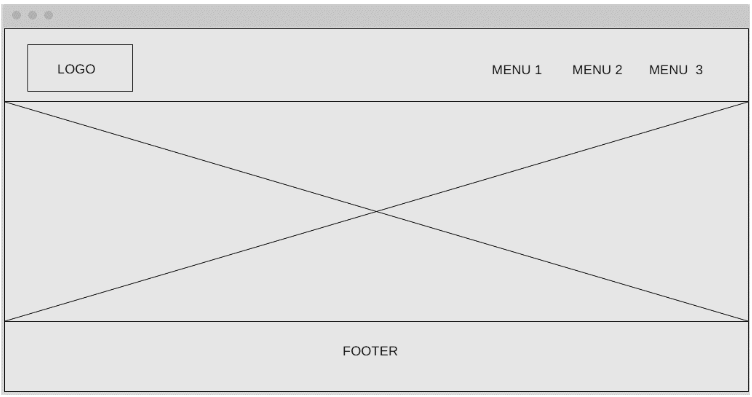

Figure 9. Homepage Interface Design

Figure 9 shows the design of the application's start page interface.

On the start page, there will be 3 main sections, namely *the header, body,* and *footer*. *The header* contains the application logo and the menus contained in the application. *The body* contains a *background image* that describes the application. *The footer* contains the identity of the application.

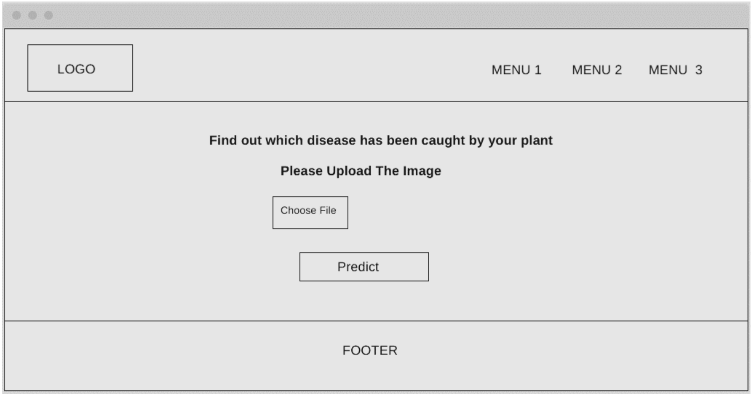

Figure 10. Image Input Process Interface Design

Figure 10 shows the interface design for the pepper plant image *input process*. This design contains an *upload button* to the system and a prediction button to start the detection process.

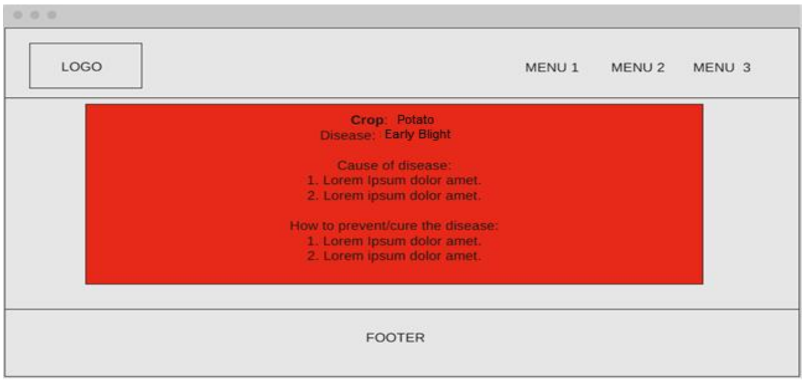

Figure 11. Prediction Results Page Interface Design

Figure 11 shows the interface design that is predicted from the input image.

4.3. Coding

At this stage, the application begins to enter the coding process. Coding is carried out in stages according to the results in the previous step. The application is developed using the web-based python programming language. The start page of the application contains the home menu, about us, and disease detection as presented in Figure 12.

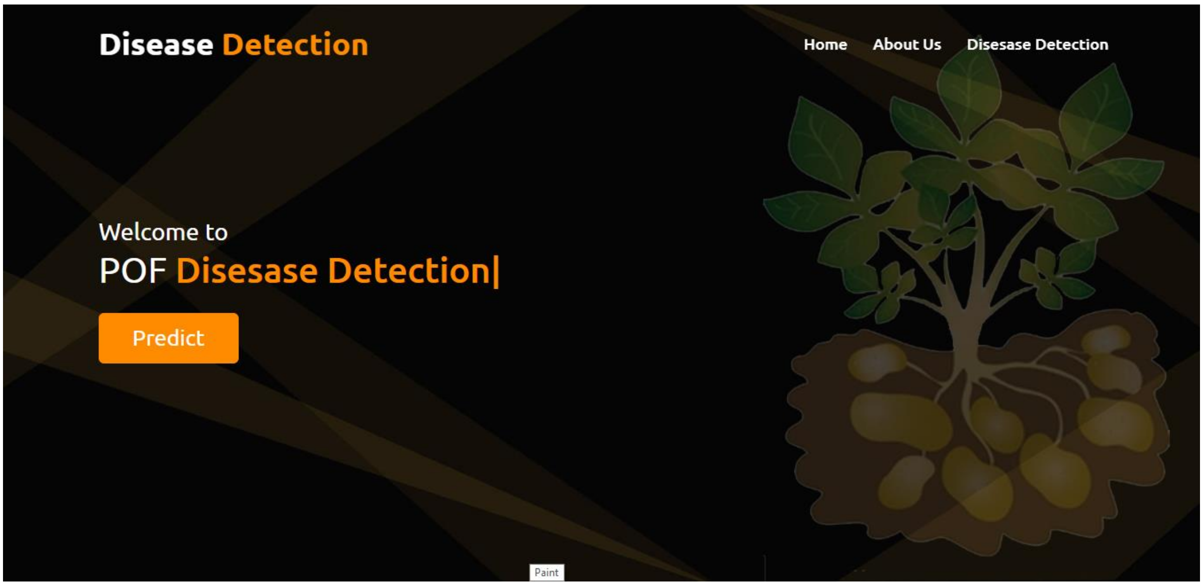

Figure 12. Homepage

The about us page contains information about the profile of Polinela Organic Farm which is the place where the research is carried out as shown in Figure 13.

Figure 13. About Us

To use the disease detection function, it can be run from the disease detection menu, the initial disease detection page will appear as shown in Figure 14.

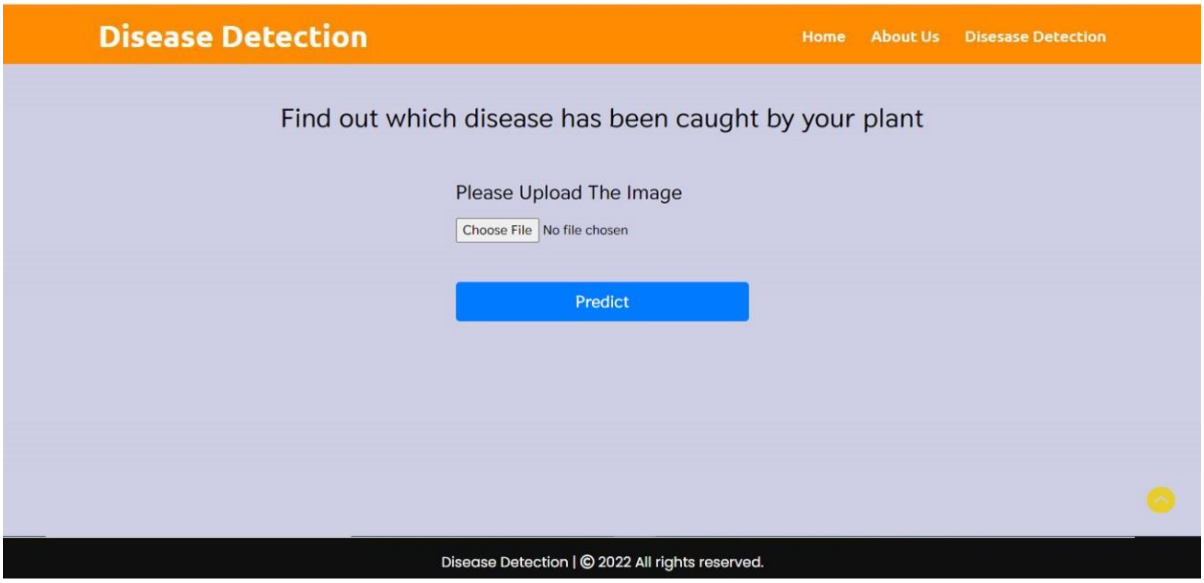

Figure 14. The start page of the disease detection menu

Then to start the detection process, the prepared leaf image data will be entered, then press the predict button, and the prediction results will appear as shown in Figure 15.

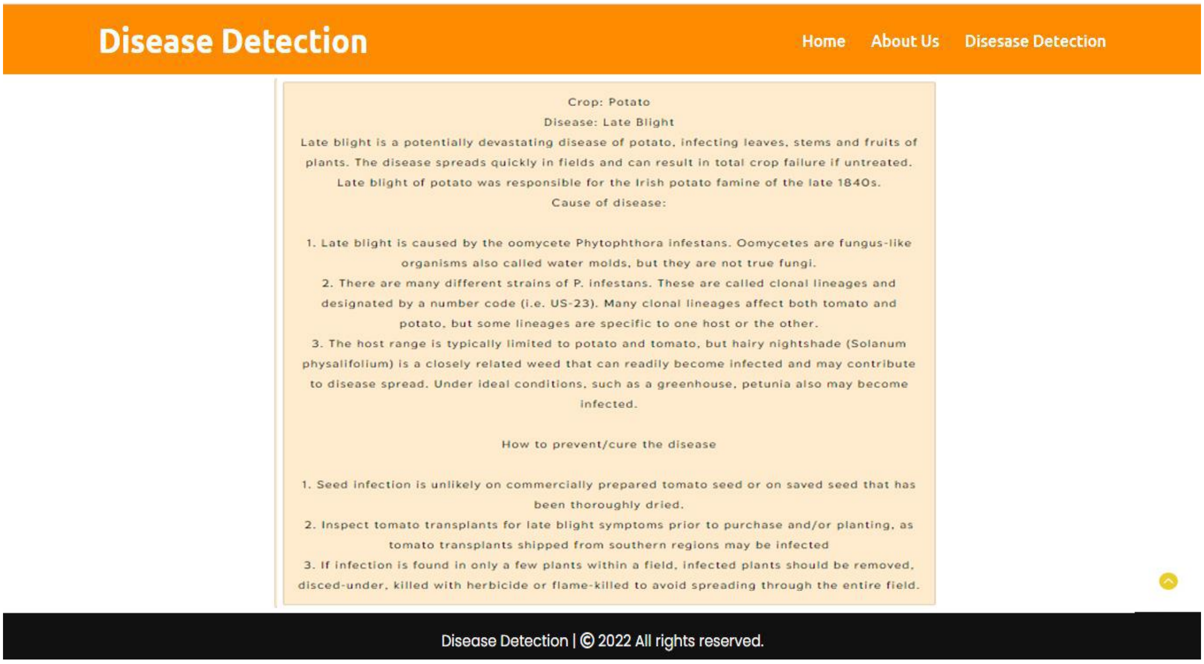

Figure 15. Disease detection results

4.4. *Testing*

This stage evaluates the application that is built. Test the application using *BlackBox testing*. The black box test tests the suitability of the interface with the desired results using the *boundary value analysis technique*. The results of the *black box* test can be seen in Table 1.

Table 1. Blackbox Test Results

No	Testing Scenario	Expected results	Test result
1	Main Page Access	Access Page Show	SUCCESS
2	Access *About Us. Page*	*About Us* Page Appears	SUCCESS
3	Access *the Disease Detection Page*	*Disease Detection* Page Shows	SUCCESS
4	Access the *disease detection process*	Show predictive results and maintenance suggestions	SUCCESS

5. Conclusions

All system functions run as expected based on the results of testing the application developed based on the black box test. This shows that the application performance is very good and can be utilized for potato farmers.

6. Acknowledgement

The authors are grateful to the Research Organization for Nuclear Energy (Project No: Rumah Program Organisasi Riset Tenaga Nuklir TR-019 Year 2022) for their kind financial support.

7. References

[1] L. K. Johnson, J. D. Bloom, R. D. Dunning, C. C. Gunter, M. D. Boyette, and N. G. Creamer, “Farmer harvest decisions and vegetable loss in primary production,” *Agric. Syst.*, vol. 176, no. October 2018, p. 102672, 2019, doi: 10.1016/j.agsy.2019.102672.

[2] L. C. Ngugi, M. Abelwahab, and M. Abo-Zahhad, “Recent advances in image processing techniques for automated leaf pest and disease recognition – A review,” *Inf. Process. Agric.*, vol. 8, no. 1, pp. 27–51, 2021, DOI: 10.1016/j.inpa.2020.04.004.

[3] S. Savalia and V. Emamian, “Cardiac arrhythmia classification by multi-layer perceptron and convolution neural networks,” *Bioengineering*, vol. 5, no. 2, 2018, DOI: 10.3390/bioengineering5020035.

[4] L. Zhang, S. Wang, and B. Liu, “Deep learning for sentiment analysis: A survey,” *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 8, no. 4, 2018, DOI: 10.1002/widm.1253.

[5] R. K. Tripathy and I. Billions, "Deep UQ: Learning deep neural network surrogate models for high dimensional uncertainty quantification," *J. Comput. Phys.*, vol. 375, pp. 565–588, 2018, DOI: 10.1016/j.jcp.2018.08.036.

[6] Y. Sun *et al.*, "Partial discharge pattern recognition of transformers based on mobile nets convolutional neural network," *Appl. Sci.*, vol. 11, no. 15, pp. 1–4, 2021, doi: 10.3390/app11156984.

[7] S. Cheon, H. Lee, C. O. Kim, and S. H. Lee, “Convolutional Neural Network for Wafer Surface Defect Classification and the Detection of Unknown Defect Class,” *IEEE Trans. Semicond. Manuf.*, vol. 32, no. 2, pp. 163–170, 2019, DOI: 10.1109/TSM.2019.2902657.

[8] A. Raza *et al.*, “A Hybrid Deep Learning-Based Approach for Brain Tumor Classification,” *Electron.*, vol. 11, no. 7, pp. 1–17, 2022, DOI: 10.3390/electronics11071146.