Vol. 14 No. 1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v14i1.8322

APPLICATION OF ARTIFICIAL NEURAL NETWORKS TO IDENTIFY DISEASES IN BROILER CHICKENS USING THE BACKPROPAGATION METHOD

Rohul Iman[1*], Abdullah Zaenuddin[2], Made Sutha Yadnaya[3]

[1,2,3]Universitas Mataram; Jl. Majapahit No.62, Gomong, Kec. Selaparang, Kota Mataram, Nusa Tenggara Barat. 83115; Phone (0370) 633007

Keywords:
Artificial Neural Network;
Backpropagation Method;
Image Data;
Matlab.

Corespondent Email:
f1b01927@student.unram.ac.id

Abstract. Artificial neural networks can be applied in the health sector. In this research, artificial neural networks were used to identify diseases in broiler chickens using the Backpropagation method. Diseases in chickens include salmonella, cocidiossis, and new castle disease. Identification of diseases in chickens based on image data from chicken droppings. The method used is backpropagation or error propagation. The data used in this research was 700 image data. The data in this research was obtained from Kaggle. The best accuracy results were obtained with a total of 600 data with 1000 iterations, and an alpha value of 0.9 obtained testing accuracy results of 90% with an error value of 10%. The accuracy results of artificial neural networks are influenced by the amount of training data. The more data used, the higher the accuracy results.

1. INTRODUCTION

Artificial neural networks are one of the information processing systems inspired by the way the human brain works in solving problems so that it can facilitate human work. Artificial neural networks are capable of learning a complex relationship between input and output quite well through supervised learning. So that it can be used to perform identification, classification, and prediction automatically [1]. Artificial neural networks are also a forecasting method that has a fairly low data error rate and performs well in the generalization process because it is supported by sufficient training data and a learning process that adjusts weights so that this model is capable of forecasting time series data for several periods in the future [2].

Backpropagation or error propagation is a common method of artificial neural network learning for completing a given task. It is a supervised learning process and is an implementation of the delta rule [3]. The advantage of Backpropagation is that it trains the network to achieve a balance between the network's ability to recognize patterns used during training and the network's ability to respond correctly to input patterns similar to those used during training. The Backpropagation method can be applied to solve problems related to identification, prediction, forecasting, pattern recognition, and so on. Its ability to learn (adaptive) and be fault tolerant makes it possible to create a robust system that works consistently [4].

Artificial Neural Network technology can be used to help identify and classify diseases that occur in broiler chickens. By using image data of chicken droppings that have been indicated as diseased, the backpropagation method is very suitable for classifying diseases in broiler chickens [5].

2. LITERATURE REVIEW

2.1. Artificial Neural Networks

Artificial neural networks are a method of grouping and separating data whose working principle is similar to that of human neural networks. Artificial neural networks are formed to solve specific problems such as pattern recognition or classification through a learning process [6].

2.3. Applications of Artificial Neural Networks

The key element of this paradigm is a new structure of the information processing system. This consists of a large number of interconnected processing elements (neurons) that work together to solve specific problems [7].

Some applications of artificial neural networks are as follows:

a. Pattern Recognition

Artificial neural networks can be used to recognize patterns (such as letters, numbers, sounds, or signatures) that have changed slightly. This is similar to the human brain, which is still able to recognize people it has not seen for some time (even if their faces or body shapes have changed slightly).

b. Signal Processing

Artificial neural networks (ADLINE model) can be used to reduce noise in telephone lines.

c. Prediction

Artificial neural networks can also be used to predict what will happen in the future based on patterns of events in the past. This is possible due to the ability of artificial networks to remember and generalize from what has happened before [8].

2.4. Network Architecture

Network architecture and training algorithms greatly determine artificial neural network models. The architecture is used to explain the direction of signal or data flow within the network. Meanwhile, the training algorithm explains how connection weights must be changed in order to achieve the desired input-output pairs [9]. Changes in connection weight values can be made in various ways, depending on the type of training algorithm used. By adjusting the weight values, it is hoped that the network's performance in learning various patterns expressed by each input-output pair will improve. Artificial neural networks have three types of architecture, namely single layer networks, multilayer networks, and recurrent networks [10].

2.5. Activation Function

In artificial neural networks, activation functions are used to normalize data to determine the output value of neurons. After activation, the resulting value will be between 0-1 but not may be 0 or 1 [10]. Some of the activation functions used are the threshold function, sigmoid function, and identity function [11].

2.6. Weight and Bias Initialization

Weights are an important part of enabling the network to generalize well to the data trained into it [12]. The choice of initial weight initialization will determine whether the network reaches a global minimum and how fast the network converges. In the network, an input unit is added whose value is always = 1. Such a unit is called a bias [13].

2.7. Backpropagation

The Backpropagation training algorithm was first formulated by Werbos and popularized by Rumelhart and McClelland for use in backpropagation. It is included in supervised learning algorithms and is typically used by multilayer perceptrons to modify the weights connected to the neurons in the hidden layer [14].

2.8. Image

An image is a representation (depiction), resemblance, or imitation of an object. Images are divided into two types: analog images and digital images. Analog images are continuous images such as those on a television monitor, X-ray photos, and so on. Meanwhile, digital images are images that can be processed by a computer. Digital images are data from images taken by a machine in the form of an approximation based on sampling and quantization. Sampling refers to the size of the squares arranged in rows and columns. In other words, sampling in an image refers to the size of the pixels (dots) in the image, and quantization refers to the brightness value expressed in grayscale according to the number of binary bits used by the machine. In other words, quantization in an image refers to the number of colors in the image [15].

3. RESEARCH METHODS

The methodology used in this study began with determining the artificial neural network workflow to obtain more accurate and expected results for chicken disease identification. The following stages used in this study can be seen in the following flowchart:

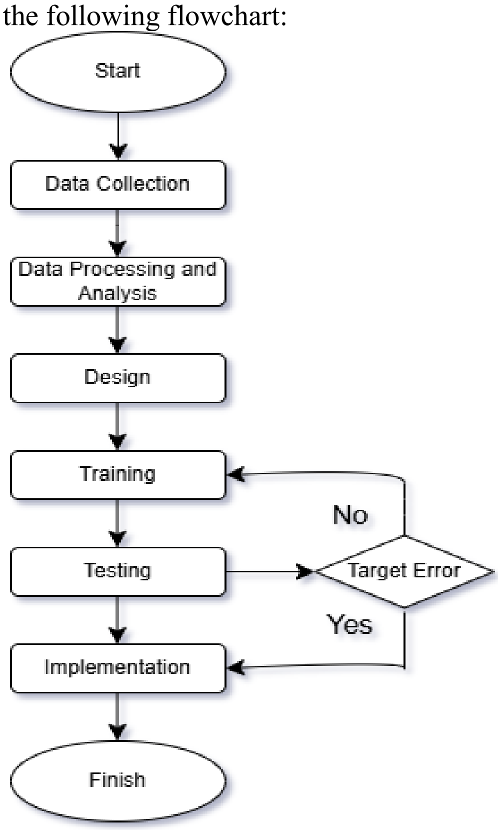


Figure 1. Research Methods

3.1. Data Collection

In the initial stage of this research, data collection is necessary to ensure its availability and completeness. The data collected consists of images and the types of diseases. In this study, the data was obtained through Kaggle. Kaggle is a website that provides many datasets. These datasets can be used to create models for projects that are being developed. This data will later be used as training data and test data in the artificial neural network system. The data used in this study consists of 400 images of chicken droppings.

Figure 2. Chicken Manure Image Data

3.2. Data Analysis and Processing

There are three main stages that must be carried out in classifying the types of diseases in chickens, namely the process of transforming colors from RGB images to HSV images, the training process using the Backpropagation artificial neural network algorithm, and the testing or classification process using the Backpropagation artificial neural network algorithm.

3.2.1 The Process of Transforming RGB Image Colors to HSV Image Colors

Color transformation from RGB to HSV is the first stage in classifying diseases in chickens. In this study, the features used in classifying image data are the average features of the H, S, and V components in the image. The following are some of the reasons why color transformation from RGB to HSV is generally performed.

The following is the equation used to perform the color transformation process from RGB to HSV.

1. Calculating Hue (H) Value:
   a. *If R, G, and B are all 0, then*
   $H = 0$ (1).
   b. *If R is the maximum value, then*
   $H = (G - B)/(max(R,G,B) - min(R,G,B))$. (2).
   c. *If G is the maximum value,* then (3).

$H = 2 + (B - R)/(max(R,G,B) - min(R,G,B))$. (3).

d. *If B is the maximum value, then* $H = 4 + (R - G)/(max(R,G,B) - min(R,G,B))$. (4).

e. *After that, H is multiplied by 60 to obtain a value on a scale of 0-360.*

2. Calculating the Saturation Value (S):
   a. $If max(R,G,B) = 0, then S = 0 (gray colored imagery)$. (5).
   b. $If max(R,G,B) \neq 0, then S = (max(R,G,B) - min(R,G,B))/max(R,G,B)$ (6).
3. Calculating the Value (V):
   a. $V = max(R,G,B)$. (7).

Before continuing the training and testing/classification process using the backpropagation neural network algorithm, the data must be translated or converted into numerical data so that it can be used for mathematical calculations. The following table shows the codes used for the Disease Type parameter data.

Disease Type	Binary Code
Healthy	11
Coccidiosis	01
Newcastle Disease	10
Salmonella	00

Figure 3. Code for Data Parameters

3.3. System Planning

At this design stage, the application design begins to take shape to determine how an application will be able to accomplish what needs to be accomplished. Basically, this design stage is a combination of the software and hardware components of a system.

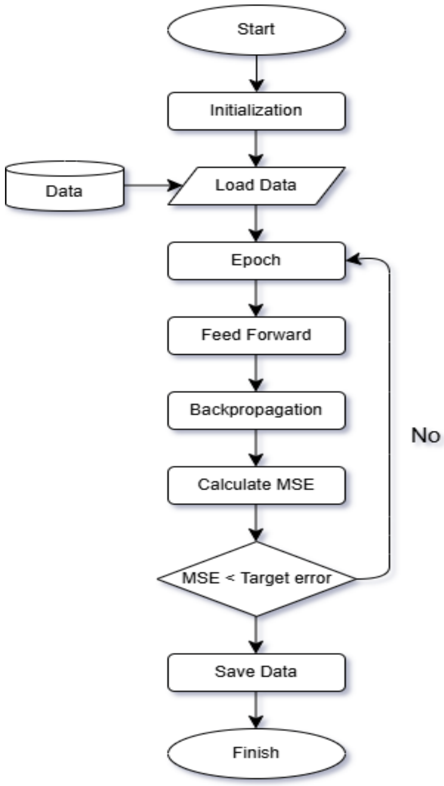


Figure 4. System Design Flowchart

3.3.1 Interface Design

Interface design is carried out using MATLAB GUI. Graphical User Interface (GUI) is a visual-oriented application program built with graphic objects as a substitute for text commands for user interaction. Graphical User Interface (GUI) in MATLAB is summarized in an application (Graphical User Interface Builder).

3.4. Training

The training process is carried out on an artificial neural network to introduce existing data patterns. The data used in the training process is the training data that was previously available, namely data on chicken disease symptoms. This training process is carried out repeatedly and uses a network architecture to find the training results with the smallest error closest to 0. The results of this training process will be influenced by the patterns used, the number of inputs, the hidden layer, the epoch, and also the learning rate. The following are the stages in the training process.

3.4.1. Artificial Neural Network Parameter Initialization

The following are the artificial neural network parameters that will be used in this case:
Number of neurons in input layer (ninput)= 3
Number of neurons in hidden layer(nhidden)= 3
Number of neurons in output layer (noutput)= 2
Learning rate (α) = 0.3
Number of iterations = 100
Error tolerance = 0.1

Based on the parameters of the number of neurons in the input layer, hidden layer, and output layer specified above, the artificial neural network architecture in this case can be described as follows.

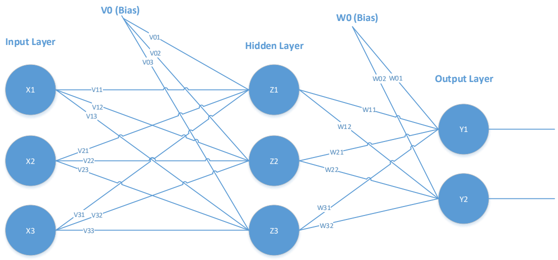


Figure 5. Architect Neural Artificial Network

3.4.2. Initializing of Initial Value of Weights V and W

The number of weights V and W is determined using the following equation:

$$weightsV = (n|input + 1) * n_{hidden} \quad (8).$$

$$weightsW = (n|hidden + 1) * n_{output} \quad (9).$$

3.4.3. Forward Propagation

1. Calculation of of neuron values in the hidden layer (Z values).

The calculation of the values of neurons Z1, Z2, and Z3 is performed using the following equation.

$$z_{\ j} = v_{0j} + \sum_{i=1}^{n} x_i v_{ij}$$

$$z_j = f\left(z_{in_j}\right)$$

$$z_{\ 1} = v_{01} + (x_1 * V_{11} + x_2 * V_{21} + x_3 * V_{31})$$

$$z_{\ 2} = v_{02} + (x_1 * V_{12} + x_2 * V_{22} + x_3 * V_{32})$$

$$z_{\ 3} = v_{03} + (x_1 * V_{13} + x_2 * V_{23} + X_3 * V_{33})$$

$$z_1 = f(z_{\ 1}) = \frac{1}{1 + e^{(-z_{\ 1})}}$$

$$z_2 = f(z_{\ 2}) = \frac{1}{1 + e^{(-z_{\ 2})}}$$

$$z_3 = f(z_{\ 3}) = \frac{1}{1+e^{(-z_{\ 3})}} \quad (10).$$

2. Calculation of neuron values in the hidden layer (Y values).

The calculation of the values of neurons Y1 and Y2 is performed using the following equation.

$$y_{\ k} = w_{0k} + \sum_{j=1}^{n} z_j w_{jk}$$

$$y_k = f\left(y_{in_k}\right)$$

$$y_{\ 1} = w_{01} + (z_1 * w_{11} + z_2 * w_{21} + z_3 * w_{31})$$

$$y_{\ 2} = w_{02} + (z_1 * w_{12} + z_2 * w_{22} + z_3 * w_{32})$$

$$y_1 = f(y\ ) = \frac{1}{1+e^{(-y\ )}}$$

$$y_2 = f(y\ ) = \frac{1}{1+e^{(-y\ )}} \quad (11)$$

3.4.4 Backward propagation

1. Calculation of backward propagation from the output layer to the hidden layer

The update of W-weight in this process is performed using the following equations.

$$\delta_k = (t_k - y_k) f'\left(y_{in_k}\right)$$

$$(t_k - y_k) y_k (1 - y_k)$$

$$\Delta w_{jk} = \alpha \delta_k z_j$$

$$\Delta w_{0k} = \alpha \delta_k$$

$$w_{jk}(baru) = w_{jk}(lama) + \Delta w_{jk}$$

$$w_{0k}(baru) = w_{0k}(lama) + \Delta w_{0k} \quad (12)$$

2. Calculation of backward propagation from the hidden layer to the input layer

The V-weight update in this process is performed using the following equations.

$$\delta_{\ j} = \sum_{k=1}^{n} \delta_k w_{jk}$$

$$\delta_j = \delta_{\ j} f'\left(z_{in_j}\right)$$

$$\delta_{\ j} z_j (1 - z_j)$$

$$\Delta V_{ij} = \alpha \delta_j x_i$$

$$\Delta V_{0j} = \alpha \delta_j$$

$$v_{ij}(baru) = v_{ij}(lama) + \Delta v_{ij}$$

$$v_{0j}(baru) = v_{0j}(lama) + \Delta v_{0j} \quad (13)$$

3.5. Testing

After training the artificial neural network and recognizing patterns, testing is performed using data different from that used for the training process. In the testing process, the backpropagation artificial neural network only performs the forward propagation process.

1. Calculation of neuron values in the hidden layer (Z values).

The calculation of the values of neurons Z1, Z2, and Z3 is performed using the following equation.

$$z_{j} = v_{0j} + \sum_{i=1}^{n} x_i v_{ij}$$

$$z_j = f\left(z_{in_j}\right)$$

$$z_{1} = v_{01} + (x_1 * V_{11} + x_2 * V_{21} + x_3 * V_{31})$$
$$z_{2} = v_{02} + (x_1 * V_{12} + x_2 * V_{22} + x_3 * V_{32})$$
$$z_{3} = v_{03} + (x_1 * V_{13} + x_2 * V_{23} + X_3 * V_{33})$$
$$z_1 = f(z_{1}) = \frac{1}{1 + e^{(-z_{1})}}$$
$$z_2 = f(z_{2}) = \frac{1}{1 + e^{(-z_{2})}}$$
$$z_3 = f(z_{3}) = \frac{1}{1 + e^{(-z_{3})}} \quad (13).$$

2. Calculation of neuron values in the hidden layer (Y value).

The calculation of the values of neurons Y1 and Y2 is performed using the following equation

$$y_{k} = w_{0k} + \sum_{j=1}^{n} z_j w_{jk}$$

$$y_k = f\left(y_{in_k}\right)$$

$$y_{1} = w_{01} + (z_1 X w_{11} + z_2 * w_{21} + z_3 * w_{31})$$
$$y_{2} = w_{02} + (z_1 * w_{12} + z_2 * w_{22} + z_3 * w_{32})$$
$$y_1 = f(y) = \frac{1}{1+e^{(-y)}}$$
$$y_2 = f(y) = \frac{1}{1+e^{(-y)}} \quad (14)$$

3.6. Calculation of Accuracy Percentage

The accuracy percentage of the backpropagation artificial neural network classification results can be calculated using the following equation.

$$presentationaccuracy(\%) = \frac{n_{true}}{n_{data}} X100$$

(15)

Where:

n_{true} : Number of classification results that are correct.

n_{data} : Number of test data

3.7. Implementation

After the design and mathematical calculations are made, the next step is to implement the entire backpropagation artificial neural network using Matlab software.

4 RESULTS AND DISCUSSION

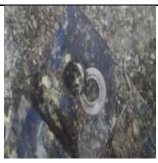
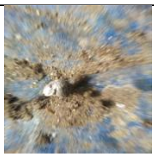
4.1. Transformation of RGB Image Colors to HSV Image Colors

To calculate the color transformation from R, G, B, to H, S, V, equations 1 -7 are used by normalizing the image intensity values by dividing each R, G, B component by 255 so that the image intensity values are in the range of 0 to 1. The following table shows the image data after conversion and the average of each H, S, V component.

are used. This is achieved by normalizing the

Table 4.1 Image data and H, S, V component values

Citra	H	S	V
	1,81	0,77	0,41
	1,16	0,43	0,44
	2,21	0,71	0,42

	1,39	0,53	0,47
	2,30	0,71	0,39
	2.18	0.71	0.40
	2.15	0.71	0.40
	2.13	0.71	0.40
	2.10	0.71	0.40
	2.08	0.71	0.40
	0,08	0,69	0,35
	0,05	0,69	0,35
	0,02	0,68	0,35
	0,06	0,68	0,42
	0.39	0,69	0,36

4.2 Artificial Neural Network Training

The training process was carried out using the data in Table 4.1. The images already had average H, S, and V values. This training included initialization of initial weights, forward propagation and backward propagation, and MSE calculation. The training output results can be seen in the following table.

Table 4.2 Target output and average error.

No	Output Target	JST Output	Average Error	Average Error^2
1	0 – 1	0.617 – 0.374	0.497	0.247
2	0 – 1	0.587 – 0.406	0.496	0.246
3	0 – 1	0.611 – 0.381	0.496	0.246
4	0 – 1	0.621– 0.372	0.496	0.246
5	1 – 0	0.357 – 0.650	0.503	0.253
6	1 – 0	0.385 – 0.622	0.503	0.253
7	1 – 0	0.411 – 0.596	0.503	0.253
8	1 – 0	0.563– 0.430	0.496	0.246
9	0 – 0	0.411 – 0.596	0.434	0.188
10	0 – 0	0.463 – 0.456	0.459	0.210
11	0 – 0	0.490– 0.586	0.488	0.238
MSE				0.165

Based on the output results and average error in Table 4.2, the MSE value from the training results in the first iteration is 0.165. The MSE value obtained from the training results in the first iteration is greater than the specified error tolerance of 0.1, so the training process for the next iteration must be carried out.

4.3 Testing the Artificial Neural Network

In the testing or classification process, the Backpropagation artificial neural network only performs a forward propagation process. In this case, the testing process calculation will be performed on test data 1.

Table 4.3 Test data to 1

X1	X2	X3	Disease Code
0,051	0,692	0,353	00

4.3.1 Calculation of the value of neurons in the hidden layer (Z-value)

$$z_{\ \ 1} = v_{01} + (x_1 X V_{11} + x_2 X V_{21} + x_3 * V_{31})$$

$$z_i n_1 = 0.1 + (0.051 * 0.2 + 0.692 * 0.3 + 0.353 * 0.1) = 0.353$$

$$z_{\ \ 2} = v_{02} + (x_1 * V_{12} + x_2 * V_{22} + x_3 * V_{32})$$

$$z_i n_2 = 0.1 + (0.051 * 0.3 + 0.692 * 0.27 + 0.353 * 0.4) = 0.443$$

$$z_{\ \ 3} = v_{03} + (x_1 * V_{13} + x_2 * V_{23} + X_3 * V_{33})$$

$$z_i n_3 = 0.1 + (0.51 * 0.1 + 0.692 * 0.25 + 0.353 * 0.35) = 0.447$$

$$z_1 = f\left(z_{\ \ 1}\right) = \frac{1}{1 + e^{\left(-z_{\ \ 1}\right)}} = \frac{1}{1 + e^{-0.353}} = 0.587$$

$$z_2 = f\left(z_{\ \ 2}\right) = \frac{1}{1 + e^{\left(-z_{\ \ 2}\right)}} = \frac{1}{1 + e^{-0.443}} = 0.608$$

$$z_3 = f\left(z_{\ \ 3}\right) = \frac{1}{1 + e^{\left(-z_{\ \ 3}\right)}} = \frac{1}{1 + e^{-0.447}} = 0.609$$

4.3.2 Calculation of the value of neurons in the output layer (Y value).

$$y_{\ \ 1} = w_{01} + (z_1 * w_{11} + z_2 * w_{21} + z_3 * w_{31})$$

$$y_i n_1 = 0.1 + (0.587 * 0.1 + 0.608 * 0.2 + 0.609 * 0.25) = 0.432$$

$$y_{\ \ 2} = w_{02} + (z_1 * w_{12} + z_2 * w_{22} + z_3 * w_{32})$$

$$y_i n_2 = 0.1 + (0.587 * 0.25 + 0.608 * 0.2 + 0.609 * 0.15) = 0.459$$

$$y_1 = f(y\ \ ) = \frac{1}{1 + e^{(-y\ \ )}} = \frac{1}{1 + e^{-0.432}} = 0.606$$

$$y_2 = f(y\ \ ) = \frac{1}{1 + e^{(-y\ \ )}} = \frac{1}{1 + e^{-0.459}} = 0.612$$

The classification process for the test data can then be performed using the same method.

The following table shows the classification results for the test data.

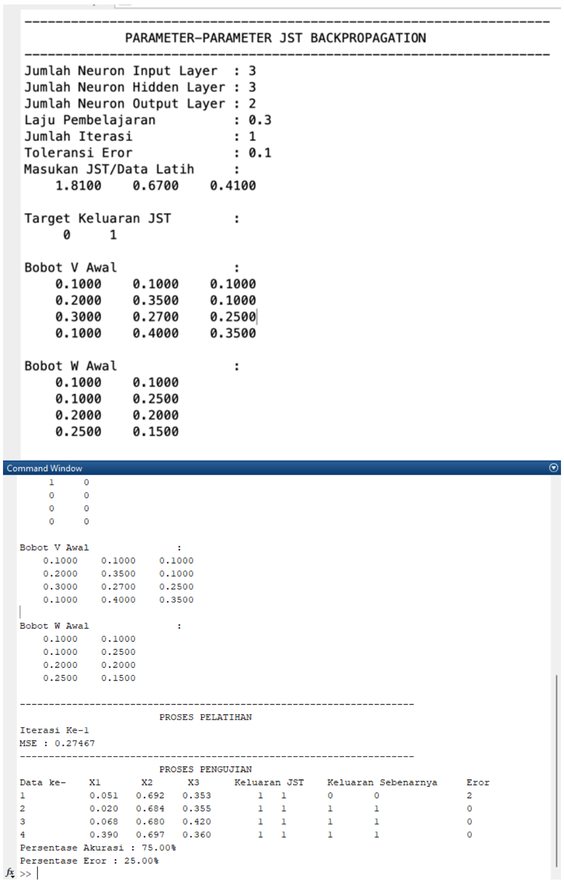
Table 4.4 Artificial neural network test results

No	ANN Output	Actual Output	Classification Result Description
1	00	00	Wrong
2	11	11	True
3	11	11	True
4	11	11	True

Based on the results of the test data qualification above, it can be seen that the number of correct outputs is 3 and the number of incorrect outputs is 1. The accuracy percentage for identifying chicken diseases in this case is 75%.

4.4. Artificial Neural Network Implementation

The artificial neural network using the backpropagation method was implemented using Matlab software. By implementing the manual calculation results in Matlab, the following results were obtained.



```
--------------------------------------------------------------------
              PARAMETER-PARAMETER JST BACKPROPAGATION
--------------------------------------------------------------------
Jumlah Neuron Input Layer  : 3
Jumlah Neuron Hidden Layer : 3
Jumlah Neuron Output Layer : 2
Laju Pembelajaran          : 0.3
Jumlah Iterasi             : 1
Toleransi Eror             : 0.1
Masukan JST/Data Latih     :
    1.8100    0.6700    0.4100

Target Keluaran JST        :
     0     1

Bobot V Awal               :
    0.1000    0.1000    0.1000
    0.2000    0.3500    0.1000
    0.3000    0.2700    0.2500
    0.1000    0.4000    0.3500

Bobot W Awal               :
    0.1000    0.1000
    0.1000    0.2500
    0.2000    0.2000
    0.2500    0.1500
```

```
Command Window
     1     0
     0     0
     0     0
     0     0

Bobot V Awal               :
    0.1000    0.1000    0.1000
    0.2000    0.3500    0.1000
    0.3000    0.2700    0.2500
    0.1000    0.4000    0.3500

Bobot W Awal               :
    0.1000    0.1000
    0.1000    0.2500
    0.2000    0.2000
    0.2500    0.1500

--------------------------------------------------------------------
                        PROSES PELATIHAN
Iterasi Ke-1
MSE : 0.27467
--------------------------------------------------------------------
                        PROSES PENGUJIAN
Data ke-    X1      X2      X3      Keluaran JST    Keluaran Sebenarnya     Eror
1           0.051   0.692   0.353       1   1       0       0               2
2           0.020   0.684   0.355       1   1       1       1               0
3           0.068   0.680   0.420       1   1       1       1               0
4           0.390   0.697   0.360       1   1       1       1               0
Persentase Akurasi : 75.00%
Persentase Eror : 25.00%
>>
```

Figure 6. Results of training and testing of artificial neural network program.

It can be seen that the results obtained in the bacpropagatioan imitation neural network program are the same as the results obtained in manual calculations. Furthermore, this artificial neural network program can be applied using Matlab GUI.

4.5 Application of the Artificial Neural Network Program Using Matlab GUI

The GUI program uses Matlab to implement the artificial neural network program that has been created. The following is the artificial neural network program code for identifying diseases in broiler chickens that has been successfully created.

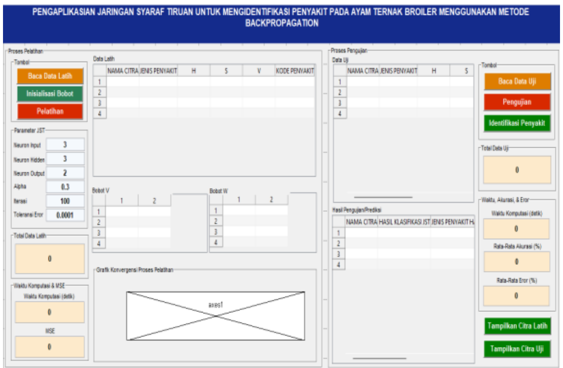


Figure 7. Artificial neural network programmer using Matlab *GUI.*

4.9 Effect of Artificial Neural Network Parameters on Accuracy

To determine the effect of artificial neural network parameters on accuracy, three different training and testing datasets were used, with the number of training and testing data being 200, 300, and 400 image data. Then, the learning rate parameter value and the number of iterations were increased. The following are the results of artificial neural network training in performing identification.

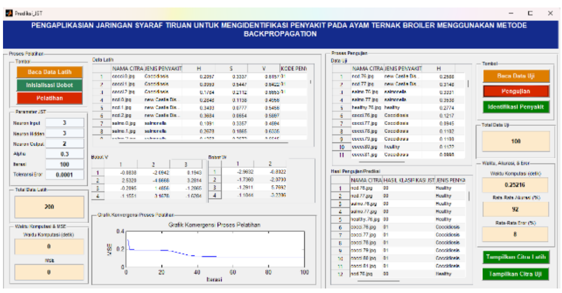


Figure 8. Prediction results for 200 training data.

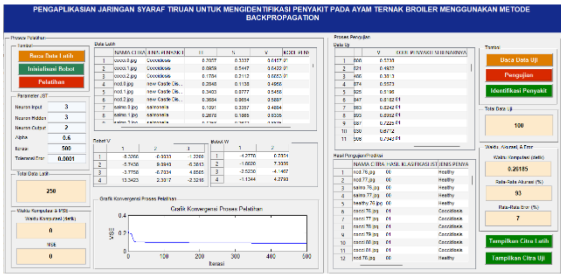


Figure 9. Prediction results for 300 training data.

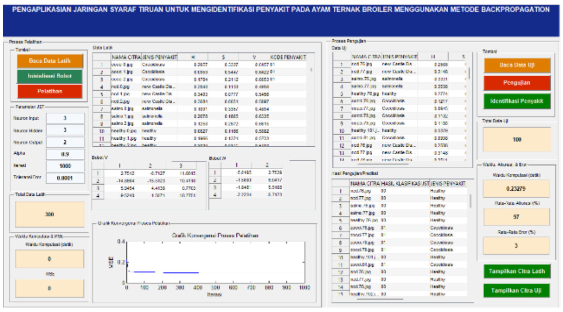


Figure 10. Prediction results for 400 training data.

Based on the training results above, it can be seen that using 200 data points, Alpha 0.3, and 100 iterations, an accuracy of 92% was obtained. With 300 data points, an Alpha value of 0.5, and 500 iterations, the accuracy was 95%. With 400 data points, an Alpha value of 0.9, and 1000 iterations, the accuracy was 97%.

The prediction results of the artificial neural network are influenced by the amount of training data used. The more data used, the higher the accuracy obtained by the artificial neural network. In addition, the values of the parameters in the artificial neural network also influence the prediction results. The smaller the error in the training process, the higher the precision will be.

4.10. Identification of Chicken Diseases Using Artificial Neural Network Backpropaftion

After completing the training and testing processes in the prediction form, the next step is the identification of chicken diseases using the backpropagation artificial neural network method. The following are the results of chicken disease identification using the backpropagation artificial neural network method.

Figure 11. Identification of chicken diseases using artificial neural networks.

5 CONCLUSION

Based on the research conducted, there are several conclusions that can be drawn, as follows.

a. It has successfully developed an artificial neural network application using the *backpropagation* method to identify diseases in broiler chickens. This application can provide information on chicken diseases using only image data or pictures of chicken droppings.

b. Based on the training conducted on the artificial neural network prediction process with *input layer* value 3, *hidden layer* value 3, *output layer* value 2, number of iterations 100, and alpha value 0.3, the test accuracy result obtained was 92% with a total of 200 training data. In the prediction process with *input layer* value 3, *hidden layer* value 3, *output layer* value 2, number of iterations 500, and alpha value 0.5, the test accuracy result was 93% with 250 training data. In the prediction process with an *input layer* value of 3, a *hidden layer* value of 3, an *output layer* value of 2, an iteration count of 1000, and an alpha value of 0.9, the test accuracy result was 97% with a test data count of 100.

c. The results of prediction and identification of *the backpropagation* artificial neural network are influenced by the amount of training data used. The more data used, the higher the accuracy obtained by the artificial neural network. In addition, the values of the parameters in the artificial neural network also influence the prediction results.

THANKS

The author would like to thank all parties who have provided support, guidance, and direction during the process of compiling and completing this research. Hopefully this research can provide benefits for the development of science and become a reference for future research.

BIBLIOGRAPHY

[1] Li. Mifeng, "Comprehensive Review of Backpropagation Neural Networks, Academic Journal of Science and Technology, 9(1), 150-154 2024.

[2] F. F. Mojtahedi, N. Yousefpour, S. H. Chow, et al., "Deep Learning for Time Series Forecasting: Review and Applications in Geotechnics and Geosciences," Archives of Computational Methods in Engineering, vol. 32, pp. 3415–3445, 2025.

[3] P. Rech, et al., "Exploiting Neural Networks Bit-Level Redundancy to Mitigate the Impact of Faults at Inference," The Journal of Supercomputing, vol. 81, p. 183, 2024.

[4] A. Kumar, et al., "Mathematical Formulation of Deep Learning Model for Poultry Disease Classification using EfficientNet-B3 CNN Model," Communications on Applied Nonlinear Analysis, vol. 31, no. 2s, 2024.

[5] D. Machuve, E. Nwankwo, N. Mduma, and J. Mbelwa, "Poultry diseases diagnostics models using deep learning," Frontiers in Artificial Intelligence, vol. 5, 2022.

[6] A. F. Pramesti and D. Suhendro, "Jaringan Saraf Tiruan untuk Memprediksi Permohonan Instalasi Listrik Menggunakan Algoritma Backpropagation," Jurnal Informatika dan Teknik Elektro Terapan (JITET), vol. 12, no. 3, pp. 315–348, 2024

[7] D. Mienye and T. G. Swart, "A Comprehensive Review of Deep Learning: Architectures, Recent Advances, and Applications," Information, vol. 15, no. 12, 2024.

[8] I. Pamungkas, S. Sumadi, and S. Alam, "Studi Komparasi Fungsi Aktivasi Sigmoid Biner, Sigmoid Bipolar dan Linear pada Jaringan Syaraf Tiruan dalam Menentukan Warna RGB Menggunakan Matlab," Jurnal Serambi Engineering, vol. VII, no. 4, pp. 3749-3758, Okt. 2022

[9] S. R. Dubey, "Activation Functions in Deep Learning: A Comprehensive Survey," Neural Networks, vol. 162, pp. 344–372, 2023.

[10] I M. Ichwan and S. F. Alfarisyi, "Pemodelan Ritme Kalori Terbakar Setiap Waktu Selama Bersepeda dengan Feedforward Neural Network," ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika, vol. 12, no. 1, pp. 13-22, Jan. 2024.

[11] K. Kurniawan, B. Ceasaro, and S. Sucipto, "Perbandingan Fungsi Aktivasi Untuk Meningkatkan Kinerja Model LSTM Dalam Prediksi Ketinggian Air Sungai Kapuas," JEPIN: Jurnal Edukasi dan Penelitian Informatika, vol. 10, no. 1, Apr. 2024.

[12] A. Saputra, S. R. Sulistiyanti, R. Marjunus, Y. Yulianti, and A. Surtono, "Penerapan Jaringan Syaraf Tiruan / JST (Backpropagation) untuk Prakiraan Cuaca di Bandar Udara Radin Inten II Lampung," Jurnal Teori dan Aplikasi Fisika, vol. 11, no. 1, Jan. 2023.

[13] L. Anas Zahir and M. Muhlis, "Jaringan Syaraf Tiruan (Neural Network) untuk Estimasi Produktivitas Pekerja Konstruksi," Jurnal DAKTILITAS, vol. 2, no. 2, pp. 142-153, Des. 2022.

[14] X. Zhang, Y. Wang, and J. Liu, "A Review of Convolutional Neural Networks in Computer Vision," Neural Computing and Applications, vol. 36, no. 14, pp. 11265–11290, 2024.

[15] W. Boulila, "An Effective Weight Initialization Method for Deep Learning: Application to Satellite Image Classification," Expert Systems with Applications, vol. 211, pp. 124344–124360, 2024.