Image Pattern Recognition in Spatial Data using Artificial Neural Network

Nursida Arif1*, Edi Nursantosa2
1Geography Education, Faculty of Social Science, Universitas Negeri Yogyakarta, Indonesia
2Directorate of Topography Army, Jakarta Pusat, Indonesia

nursida.arif@uny.ac.id

Abstract. This study predicts erosion based on the image patterns as the input data by using an ANN approach. Several simulations had been carried out to get the ANN parameter combination in producing the best accuracy through trials and errors. The results show that the accuracy of artificial neural network training is not influenced by the number of channels, namely the input dataset (erosion factors) and the dimensions of the data, but it is determined by changes in the network parameters. The best combination of parameters is 2 hidden layers, learning rate 0.001, Momentum 0.9, and RMS 0.0001 with an accuracy of 98.55%

1. Introduction
The development of remote sensing technology from aerial photography to satellite imagery in the 1980s makes the pixel-based spatial analysis widely used. Remote sensing has become a solution in the acquisition of spatial data, which is fast and efficient but requires proper analytical techniques, if it is integrated with non-spectral spatial data such as measurements in the field. At present, the development of studies on spatial data analysis techniques is increasing along with the technological advances that demand high accuracy and time efficiency both in the analysis and data collection. The pixel-based spatial data analysis is strongly influenced by the classification method used because each method has a different treatment for the input data [1], [2]. The classification of input data that relies on spectral can be done using conventional methods based on statistical principles such as maximum likelihood where the data are normally distributed [3]. The spatial data with mixed or combined spectral characteristics between spectral and non-spectral data require analysis techniques with machine learning including decision trees [4], [5], [6], [7], fuzzy logic [8], [9], [10], and artificial neural networks [2], [11], [12], [13].

Artificial Neural Network is an effective method with higher accuracy than that of the decision tree [14], while fuzzy logic is more appropriate to be used for data with probabilistic membership of a vague class [9]. Artificial neural network is an information processing system that mimics the work of the neural network system in the human brain that represents knowledge with learning ability [15]. The main problems of the spatial data processing are large data dimensions and data uncertainties [16], [17]. ANN can easily understand uncertain data through network training and the right combination of input parameters [12]. ANN is widely used for an image analysis including filtering, interpretation, and prediction. In this study, ANN was used to predict the level of erosion through pattern recognition of several input data combinations.
The sample was taken from each input data as the training data on ANN representing each class of erosion. Input data are a combination of spectral images, namely SPOT imagery and non-spectral images. SPOT imagery is as the data source of vegetation cover data, while non-spectral images are obtained from the interpolation of data from the measurements in the field. In training data, the ANN architecture used is Multilayer Perceptron (MLP) with backpropagation algorithm. Some previous researchers proved that ANN with MLP is able to provide accurate predictions in cases with complex data [1], [18]. Backpropagation is a supervised learning algorithm and is usually used by MLP to change the weights associated with neurons in the hidden layer [15].

The aim of using backpropagation is to obtain a balance between correct training pattern recognition and good responses to other similar patterns (testing data). The network can be trained continuously until all training patterns are correctly identified. In this study, a testing was conducted through simulations of various combinations of ANN parameters to find the best accuracy in recognizing patterns. The ANN parameters include iteration, learning rate, momentum, hidden layer, and RMS error. Iteration (I) is the number of repetitions performed during data training. [19] produces better accuracy (> 60%) in iterations of 10000 to 25000. Learning rate (LR) is a constant speed of learning, where the selection of the right constant will speed up the network in learning to recognize patterns, so that only little repetition occurs [15], [20]. Momentum (M) is a network parameter that serves to accelerate convergence and prevent local optimum, with values ranging from 0 to 1. Hidden layer is the layer that receives responses in the form of weights from the input layer to be forwarded to the output layer. RMS error is the value used to determine the RMS value limit, so that the iteration is stopped. Iteration will stop if the RMS value is lower than the tolerance limit value set on the network. Basically, there are no provisions on the exact value of each network parameter to obtain high accuracy, because ANN is based on "trial and error". Therefore, it is necessary to do a lot of simulations to test the best combination of the parameter values. The simulation results become a reference for analyzing the ability of the network in the recognition of erosion spatial data.

2. Methods

2.1. ANN-Multi Layer Perceptron

Information processing on ANN is divided into three, namely the input layer, hidden layer, and the output layer. Information will be collected in layers called neuron layers. Information given to ANN will be propagated from layer to layer, starting from the input layer to the output layer through another layer, which is often known as the hidden layer.

The algorithm used in this study is backpropagation. Farrokhzad [21] made an ANN-based modeling with backpropagation algorithm to analyze certain elements causing landslides in the study area and soil profiles. The results of the modeling accuracy with ANN were compared to data which were not used in the ANN training in accordance with the analysis of field data; the accuracy of ANN training obtains > 93%.

The backpropagation algorithm uses an error output to change its weight values in the backward direction. To get this error, the forward propagation stage must be done first. At the time of forward propagation, neurons are activated by using an activation function that can be differentiated [15]. The activation function used in this study is the sigmoid function:

\[ y = f(x) = \frac{1}{1 + e^{-\sigma x}} \]  
\[ f'(x) = \sigma f(x)[1-f(x)] \]  

Backpropagation Algorithm

- Initialization of weights (weights are taken with a fairly small random value)
- Set: iteration, target error, and learning rate (\( \alpha \))
Feedforward

a. Each input unit (Xi, i = 1,2,3, ..., n) receives the xi signal and forwards the signal to all units in the layer above it (hidden layer)

b. Each unit in the hidden layer (Zj, j = 1,2,3, ..., p) sums the weighted input signals:

\[ z_{\text{in}}^j = b_1^j + \sum_{i=1}^{n} x_i v_{ij} \]  

(3)

use the activation function to calculate the output signal:

\[ z_j = f(z_{\text{in}}^j) \]  

(4)

and send the signal to all the units in the upper layer (output units)

\[ z_j = f(z_{\text{in}}^j) \]  

(4)

c. Each unit of output (Yk, k = 1,2,3, ..., m) sums the weighted input signals.

\[ z_{\text{in}}^k = b_2^k + \sum_{j=1}^{p} z_j w_{jk} \]  

(5)

Use the activation function to calculate the output signal:

\[ y_k = f(y_{\text{in}}^k) \]  

(2.7)

and send the signal to all the units in the upper layer (output units)

note:
Step (b) is done as many as the number of hidden layers

Backpropagation

1. Each unit of output (Yk, k = 1,2,3,....,m) receives a target pattern related to the learning input pattern. Calculate the error information:

\[ \delta_2^k = (t_k - y_k) f'(y_{\text{in}}^k) \]  

(6)

\[ \varphi_2^k = \delta_k z_j \]  

(7)

\[ \beta_2^k = \delta_k \]  

(8)

Then calculate the weight correction (which will be used to correct the w_{jk} value):

\[ \Delta w_{jk} = \alpha \varphi_2^k \]  

(9)

Also calculate the correction bias (which will later be used to correct the b_2^k value):

\[ \Delta b_2^k = \alpha \beta_2^k \]  

(10)

Step (a) is also done as many as the number of hidden layers, i.e. calculating error information from a hidden layer to the previous layer.

2. Each hidden unit (Zj, j = 1,2,3, ..., p) adds up the input delta (from the units in the upper layer):

\[ \delta_{\text{in}}^j = \sum_{k=1}^{m} \delta_k w_{jk} \]  

(11)

multiply the value by \( \delta_{\text{in}}^j \) with the derivative of the activation function to calculate the error information:

\[ \delta_1^j = \delta_{\text{in}}^j f'(z_{\text{in}}^j) \]  

(12)

\[ \varphi_1^j = \delta_1^j x_j \]  

(13)

\[ \beta_1^j = \delta_1^j \]  

(14)

calculate the weight correction (which will be used later to correct the v_{ij} value):

\[ \Delta v_{ij} = \alpha \varphi_1^j \]  

(15)

calculate the bias correction (which will later be used to correct the b_1^j value):

\[ \Delta b_1^j = \alpha \varphi_1^j \]  

(16)

3. Each unit of output (Yk, k = 1,2,3, ..., m) corrects the bias and weight (j = 0,1,2,...,p):

\[ w_{jk}(\text{newer}) = w_{jk}(\text{older}) + \Delta w_{jk} \]  

(17)

\[ b_2^k(\text{newer}) = b_2^k(\text{older}) + \Delta b_2^k \]  

(18)

Each hidden unit (Zj, j = 1,2,3, ..., p) fixes its bias and weight (i = 0,1,2,...,n):

\[ v_{ij}(\text{newer}) = v_{ij}(\text{older}) + \Delta v_{ij} \]  

(19)
\[ b_1(\text{newer}) = b_1(\text{older}) + \Delta b_1 \]

(20)

2.2. Input Dataset

The erosion control factor becomes the input and output layers, namely the class of erosion (very slight, slight, moderate, severe, and very severe). Artificial neural networks require two stages of analysis, namely stages of data training and data testing. The overview of the inputs and outputs that occur in this study is presented in Figure 1.

The sample area is taken from each erosion class with a total pixel number of 2216 (Table 1) consisting of 30 samples. The determination of the sample area is based on the results of interpretations on SPOT 5 satellite imagery and the results of field analysis. There are 9 input images, consisting of 4 channels in Spot Image 5, Image R as the representations of climate factors, Image K as the representations of soil factors, Image LS as the representations of topographic factors, Image C as the representations of vegetation factors, and Image P as the representation of land management factors. Each input image is presented in Figure 2.

![Figure 1. ANN structure of Erosion Model](http://www.beak.de/advangeo/advangeo_prediction)

The sample area is taken from each erosion class with a total pixel number of 2216 (Table 1) consisting of 30 samples. The determination of the sample area is based on the results of interpretations on SPOT 5 satellite imagery and the results of field analysis. There are 9 input images, consisting of 4 channels in Spot Image 5, Image R as the representations of climate factors, Image K as the representations of soil factors, Image LS as the representations of topographic factors, Image C as the representations of vegetation factors, and Image P as the representation of land management factors. Each input image is presented in Figure 2.

### Table 1. Number of Training Data Pixels

| Erosion Class      | Total Pixel | Total (%) |
|--------------------|-------------|-----------|
| Very slight        | 162         | 7.31      |
| Slight             | 295         | 13.31     |
| Moderate           | 359         | 16.20     |
| Severe             | 425         | 19.17     |
| Very severe        | 975         | 43.99     |
| **Total**          | **2216**    | **100**   |
B1, B2, B3, B4 = SPOT Image Channels; R = climate factor; K = soil factor; LS = topography factor; C = vegetation factor; P = land management factor

**Figure 2.** Input images of ANN Model
The mathematical calculation of ANN from the image input data above uses the backpropagation algorithm done by following the illustration in Figure 3 where the ANN parameters used refer to Table 1.

![Figure 3. Architecture of ANN Model](image)

**Table 2. Network Parameters**

| Network Parameters | Values |
|--------------------|--------|
| Hidden layer       | 1 and 2 [15] |
| Learning rate      | 0.001, 0.01 [2] |
|                    | 0.4 [22] |
| Momentum           | 0.4, 0.5 and 0.9 [1], [23] |
| RMS                | 0.001; 0.0001 [1], [2] |

3. Results and Discussion

The recognition of input data patterns with ANN is done using 23 trials and error simulations with the different levels of accuracy as presented in Table 2. Arif and Danoedoro [2] conducted 20 ANN simulations to get the best classification results.

**Table 3. ANN Simulation and Accuracy**

| Simulation | HL | LR | M     | RMS  | I   | Overall accuracy |
|------------|----|----|-------|------|-----|------------------|
| ANN 1      | 1  | 0.001 | 0.5  | 0.0001 | 5000 | 11.68            |
| ANN 2      | 1  | 0.001 | 0.5  | 0.0001 | 10000| 62.54            |
| ANN 3      | 1  | 0.001 | 0.5  | 0.0001 | 15000| 85.65            |
| ANN 4      | 2  | 0.001 | 0.5  | 0.0001 | 5000 | error            |
| ANN 5      | 2  | 0.001 | 0.5  | 0.0001 | 15000| error            |
| ANN 6      | 1  | 0.001 | 0.9  | 0.0001 | 15000| 97.65            |
| ANN 7      | 2  | 0.4   | 0.5  | 0.0001 | 5000 | error            |
| ANN 8      | 1  | 0.01  | 0.5  | 0.001  | 5000 | 91.96            |
| ANN 9      | 1  | 0.01  | 0.5  | 0.001  | 10000| 90.7             |
| ANN 10     | 2  | 0.01  | 0.5  | 0.001  | 5000 | 45.39            |
| ANN 11     | 2  | 0.01  | 0.5  | 0.001  | 10000| 92.28            |
Table 2 shows that the lowest accuracy occurs in ANN 4, ANN 5, and ANN 7 simulations. ANN 1 simulation only produces an accuracy of 11.68%; in ANN 4 the same parameters are tried using 2 HL, in which the results cannot be executed by the network (error) as happened in ANN 5 and ANN 7 simulations. The examples of outputs produced under error conditions are that among the 30 trained patterns, ANN usually only recognizes two patterns. The parameter on ANN 4 was tried on ANN 5 and ANN 19 by adding the number of iterations; low results were still obtained. This means that the addition of iteration to the momentum values of 0.5, 2 HL, LR 0.001 and RMS 0.0001 is less optimal. Most of the low accuracy occurs in the simulations with 2 HL. HL is used to analyze the impact of ANN training iterations [24]. However, if the simulation with HL 1 produces low accuracy, the number of iterations as in ANN 1 is added with the same parameters (momentum, HL, LR, RMS); the number of iterations is added to 10000 on ANN 2 which results in a higher accuracy of 62.54%. The results of the 23 simulations carried out based on the trials and errors obtain several simulations with an accuracy of >80% as presented in Figure 4.
Figure 4 shows several classification results with different patterns. Figure 4(e) is a classification result with higher accuracy than the overall simulation that has been done. The highest accuracy reaches 98.55% with 2 HL network parameters; LR 0.001; M 0.9; RMS 0.0001; iteration of 20000. The results were higher than those produced by [24], which reached 95% in the land use classification. The accuracy of ANN training is influenced by the input dataset (erosion factors) and changes in the network parameters [1]. These findings provide an opportunity for future research to test other ANN algorithms and validation methods for pattern recognition results to determine the most optimal method.

Acknowledgements
The authors would like to thank National Institute of Aeronautics and Space, Indonesia and Meteorological Climate and Geophysics Agency for providing the data.

References
[1] Arif N Danoedoro P and Hartono, 2017 Analysis of Artificial Neural Network in Erosion Modeling: A Case Study of Serang Watershed in IOP Conference Series: Earth and Environmental Science 98, 1.
[2] Arif N and Danoedoro P, 2013 An analyze of a backpropagation neural network in the identification of critical land based on ALOS imagery in 34th Asian Conference on Remote Sensing 2013, ACRS 2013 1.
[3] Shivakumar B R and Rajashekararadhya S V., 2018 Investigation on land cover mapping capability of maximum likelihood classifier: A case study on North Canara, India Procedia Comput. Sci. 143 p. 579–586.
[4] Amirruddin A D Muharam F M Ismail M H Ismail M F Tan N P and Karam D S, 2020 Hyperspectral remote sensing for assessment of chlorophyll sufficiency levels in mature oil palm (Elaeis guineensis) based on frond numbers: Analysis of decision tree and random forest Comput. Electron. Agric. 169, May 2019 p. 105221.
[5] Ghulam A Porton I and Freeman K, 2014 Detecting subcanopy invasive plant species in tropical rainforest by integrating optical and microwave (InSAR/PolInSAR) remote sensing data, and a decision tree algorithm ISPRS J. Photogramm. Remote Sens. 88 p. 174–192.
[6] Otukei J R and Blaschke T, 2010 Land cover change assessment using decision trees, support vector machines and maximum likelihood classification algorithms Int. J. Appl. Earth Obs. Geoinf. 12, SUPPL. 1 p. 27–31.
[7] Yang C C et al., 2003 Application of decision tree technology for image classification using remote sensing data Agric. Syst. 76, 3 p. 1101–1117.
Ahmad I et al., 2020 GIS and fuzzy logic techniques-based demarcation of groundwater potential zones: A case study from Jemma River basin, Ethiopia. *J. African Earth Sci.* **169**, April p. 103860.

Saadoud D Hassani M Martin Peinado F J and Guettouche M S, 2018 Application of fuzzy logic approach for wind erosion hazard mapping in Laghouat region (Algeria) using remote sensing and GIS. *Aeolian Res.* **32**, January p. 24–34.

Silver M Svoray T Karnieli A and Fredj E, 2020 Improving weather radar precipitation maps: A fuzzy logic approach *Atmos. Res.* **234**, June 2019 p. 104710.

Gholami V Booij M J Nikzad Tehrani E and Hadian M A, 2018 Spatial soil erosion estimation using an artificial neural network (ANN) and field plot data *Catena*.

Jahangir M H Mousavi Reineh S M and Abolghasemi M, 2019 Spatial predication of flood zonation mapping in Kan River Basin, Iran, using artificial neural network algorithm *Weather Clim. Extrem.* **25**, June p. 100215.

Philippopoulos K and Deligiorgi D, 2012 Application of artificial neural networks for the spatial estimation of wind speed in a coastal region with complex topography *Renew. Energy*.

Xiong Y Z Wang R and Li Z, 2010 Extracting land use/cover of mountainous area from remote sensing images using artificial neural network and decision tree classifications: A case study of Meizhou, China *Proc. - 2010 Int. Symp. Intell. Inf. Process. Trust. Comput. IPTC 2010* November p. 133–136.

Fausett L, 2005 Fundamentals of Neural Networks: Architectures, Algorithms and Applications *J. Chromatogr. B. Analyt. Technol. Biomed. Life Sci.* **814**, 2 p. 315–323.

Li L Ban H Wechsler S P and Xu B, 2017 Spatial Data Uncertainty *Compr. Geogr. Inf. Syst.* **3** p. 313–340.

Pogson M and Smith P, 2015 Effect of spatial data resolution on uncertainty *Environ. Model. Softw.* **63** p. 87–96.

Yang S Feng Q Liang T Liu B Zhang W and Xie H, 2018 Modeling grassland above-ground biomass based on artificial neural network and remote sensing in the Three-River Headwaters Region *Remote Sens. Environ.*

Arif N and Danoedoro P, 2014 Study on Experiments of Artificial Neural Network Using Spatial Data *4*, 8 p. 732–736.

Mollalo A Mao L Rashidi P and Glass G E, 2019 A gis-based artificial neural network model for spatial distribution of tuberculosis across the continental united states *Int. J. Environ. Res. Public Health* **16**, 1 p. 1–17.

Farrokhzad F Barari A Choobbasti A J and Ibsen L B, 2011 Neural network-based model for landslide susceptibility and soil longitudinal profile analyses: Two case studies *J. African Earth Sci.* **61**, 5 p. 349–357.

Chen G and Zhai M, 2019 Quality assessment on remote sensing image based on neural networks *J. Vis. Commun. Image Represent.*

Song H Xu R Ma Y and Li G, 2013 Classification of ETM+ remote sensing image based on hybrid algorithm of genetic algorithm and back propagation neural network *Math. Probl. Eng.* **2013**.

Abbas Q Haider Bangyal W and Ahmad J, 2013 The Impact of Training Iterations on ANN Applications Using BPNN Algorithm *Int. J. Futur. Comput. Commun.* **2**, 6 p. 567–569.