Retinal Diseases Classification Using Levenberg-Marquath (LM) Learning Algorithm for Pi Sigma Network (PSN) and Principal Component Analysis (PCA) Methods

S Swasono¹, A Damayanti², and A B Pratiwi³
Mathematics Department of Science and Technology Faculty of Universitas Airlangga, Indonesia

E-mail: ²aulid@fst.unair.ac.id

Abstract. Eye is one of the most important parts of the body that has many parts or layers, one of which is the retina of the eye. Retina is a slight layer coating the back of the eye and serves to receive light, converting light into nerve signals which then sent to the brain. As well as the other body parts, retina is also able of experiencing disorders or abnormalities that can inhibit the vision process. Several of the retinal eye diseases are Retinal Ablatio, Age-related Macular Degeneration, and Diabetic Retinopathy. Eye fundus examination can be used to classify eye retinal diseases. Pi Sigma Network with Lavenberg Marquath Learning Algorithm is a method applied for classification of eye retinal diseases. Before executing classification process, image processing resulting grayscale image and data reduction using Principal Components Analysis are carried out for reducing the size of the fundus image. The accuracy obtained from the classification process using the applied methods is 100%.

1. Introduction
Eye is one of the most crucial body parts that detects light and changes it into impulses at nerve cell. Part of the eye that is sensitive to light stimulation is retina, which is located at the back of the eye. This means that the role of retina is very crucial in vision system. Any kind of abnormality should be responded quickly in order to avoid any more damage that can disrupt vision system, whether partly or whole. There are several retinal abnormalities, for instance, Age-related Macular Degeneration (AMD), Retinal Detachment (RD), and Diabetic Retinopathy (DR).

One of the ways to differentiate or classify these retinal disruptions is by using neural network, in this case, we use Pi Sigma Network (PSN). PSN was first introduced by Shin and Ghosh in 1991. The architecture of PSN shows that the weights adapting during the learning process of training algorithm occurs only to the weights that connecting the input neurons of input layer to the hidden neurons of hidden layer, while the weights that connecting the hidden neurons to output layer are fixed to one. Due to this such of topology, it helps on reducing the time for training algorithm [1]. In addition, PSN provides higher order capabilities in increasing its computational power, provides better capability in classifying [2], and requires less memory [3].

The general structure of PSN uses Gradient Decent Algorithm (GDA) for the weights adapting in training algorithm [3]. But in this study, the training algorithm uses the Levenberg-Marquardt (LM) Learning Algorithm instead since it can simplify the optimization of Newton algorithm resulting faster...
convergence if it is compared to basic GDA method. It is also not as complex as Newton’s optimization algorithm [4].

As we use fundus image of retina as the data of the whole process, we will need a method to reduce the value of the image without losing much of its information, so we use Principal Components Analysis (PCA). PCA is a mathematical equation that is used in data dimensions reducing and allows standards identification of data and its expression so that its similarities and differences are emphasized. When the patterns are found, compressing can be done. PCA can also be used as compression algorithm of image with a low level of information loss [5].

In this study, we will use various learning rate value for each trial on classifying the fundus images, resulting in various percentage of training and validation process. In the previous research [6] which classifying Diabetic Retinopathy (DR) using fundus images has a result that the classification accuracy using Artificial Neural Network (ANN) and Local Binary Pattern (LBP) is 96.73%. Another research [7], which classifying Diabetic Retinopathy (DR) and Age-related Macular Degeneration (AMD) using Support Vector Machine (SVM) can achieve 96.67% of classification accuracy. This study is expected to have a better result than the previous studies in classifying retinal diseases using the Pi Sigma Network (PSN) and Principal Component Analysis (PCA) methods.

2. Methodology

The stages in this study are pre-processing, feature reduction, and retinal diseases classification.

2.1. Pre-processing

Pre-processing is the beginning of data input processing before getting into the main stage, PSN process. In this study, the stage of pre-processing is done by turning the original image into grayscale image. The purpose of this stage is to improve the quality of the image in order to give clearer and more detail information about the image’s characteristic, numerically [8].

2.1.1. Grayscale Image

Grayscale image is an image which has one channel value for each of the pixel. Grayscale can be represented in two-dimension array which each of its element shows the intensity of the image. Grayscale can be calculated using the following equation:

\[ \text{Grayscale} = \frac{\text{red} + \text{green} + \text{blue}}{3} \]  

(1)

Grayscale value is in the range of 0 to 255 which 0 represents the most obscure intensity and 255 represents the brightest intensity [8].

2.2. Principal Components Analysis (PCA)

PCA reduces data dimensions by using linear combinations for its principal components. If there are \( p \) variables, then there will be \( p \) principal components. The result of PCA can be obtained by analyzing in the process of finding the eigenvalues of a sample covariance matrix. The steps normally followed in a PCA of a digital image can be established by following the steps below [5]:

**Step 1:** Correcting the image so that its columns have zero means and unitary variances.

\[ CI = I - MI \]  

(2)

**Step 2:** Calculating the C matrix covariance using this following equation:

\[ C = CI \times (CI)^T \]  

(3)

**Step 3:** Calculating the eigenvalues and the corresponding eigenvectors.

**Step 4:** The value of the characteristic vector \((v)\) is obtained in shape of a matrix containing the list of eigenvectors (matrix columns) of the covariance matrix.

**Step 5:** Calculating the final data

\[ f = vc^T \times (I - MI)^T \]  

(4)

**Step 6:** The matrix of original image is obtained from the final data without compression using
$$I^T = (vc)^T \times f + MC^T \quad (5)$$

**Step 7:** Any components explaining only a small compression are discarded.

Where $I$ = image corrected by the mean

$L$ = image

$M$ = mean of the image

$C$ = covariance of the image

$f$ = matrix of the covariance’s eigenvectors

### 2.3. Pi Sigma Network (PSN)

![Figure 1. Structure of PSN](image)

The figure above shows the structure of PSN. The input layer contains a vector of $n$ input and $x_k$ is the $k$-th node of input layer. These inputs are connected to the hidden unit of $K$ nodes by the adapted weight $w_k$. Let $h_j$ be the value of $j$-th node of the hidden layer and $y_i$ is the value for the $i$-th pattern output of PSN, $\theta_j$ is an adjustable threshold, then,

$$h_j = \sum_{k=1}^{N} w_{ji} x_k + \theta_j \quad (6)$$

$$y_i = \alpha \left( \prod_{j=1}^{K} h_j \right) \quad (7)$$

$\sigma(x)$ is an activation function, and is selected as the binary function.

$$\sigma(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \quad (8)$$

Because the output is the product of all the hidden nodes, there is no need in updating all the weights. We can randomly choose a hidden node and update its adjustable weights using Gradient Decent Algorithm (GDA).

$$\Delta \theta_j = \alpha (d_i - y_i) y_i \prod_{j \neq i}^{K} h_j \quad (9)$$

$$\Delta w_{kli} = \alpha (d_i - y_i) y_i \prod_{j \neq i}^{K} h_j x_k = \Delta \theta_j x_k \quad (10)$$

Where $\alpha$ is learning rate value, $p$ is the $p$-th training pattern, $d_i$ and $y_i$ are the target and actual output, $\sigma(x)$ is the first derivative of activation function $\sigma(x)$. Below is the formulation to calculate the MSE.
\[ e^2 = \frac{1}{2} \sum_{p} \sum_{i} (t_i^p - y_i^p)^2 \]  

(11)

2.4. Levenberg-Marquardt Learning Algorithm

Levenberg-Marquardt (LM) Learning Algorithm is developed from backpropagation algorithm using hessian matrix approach to determine the changes of its weight and bias (Damayanti, 2016) and is designed to minimize the function of sum-of-square error (Aldrich, 2002). Weight and bias changes can be calculated using these formulations below.

\[ H = J^T e \]  

(12)

\[ g = J^T J \]  

(13)

\[ \Delta X = [J^T J + \alpha I] J^T e \]  

(14)

\[ X = X + \Delta X \]  

(15)

where  

- \( X \) = weight and bias function  
- \( e \) = vector that states all error in the network output  
- \( \alpha \) = learning rate  
- \( I \) = identity matrix

3. Data Sets and Results

The dataset that are used in this study consists of AMD, DR, normal retina, and RD fundus images that are obtained from Rumah Sakit Mata Masyarakat Surabaya. Each category has 15 images, so the total images data are 60 fundus images. The amount of images have shown a good result for this research. These images are divided into two different processes, 40 fundus images for training process and another 20 fundus images are for validation process with the type is .jpg and pixel size are 185 × 185 pixels. The figure below shows the fundus image of each category.

Figure 2. Fundus images of

(a) RD   (b) AMD   (c) Normal Retina   (d) DR

There are several stages done in this study, that are pre-processing, feature reduction with PCA, and retinal diseases classification using PSN and LM Learning Algorithm. In image pre-processing, the original fundus images will be turned into grayscale image. Figure 2 shows the result of changing original image into grayscale image.

Figure 3. Fundus images

(a) Original Image   (b) Grayscale Image
After going through image pre-processing, the next stage is feature reduction using PCA. The input for PCA reduction uses the numerical result of pre-processing containing a matrix with the size is $34225 \times 60$. This size will get into PCA until the size is reduced to $60 \times 60$, called the principal component matrix ($F$). Then, it will be normalized in order to reduce the value of $F$ in between $-1$ and $1$.

The classification step using PSN and LM Learning Algorithm begins with inputting $P$ and initializing the parameters needed, such as learning rate ($\alpha$), maximum iteration, and MSE target. It is then followed by feedforwarding process to hidden layer and output layer. After the output is done, then calculating the output value. The process of weights updating using LM Learning Algorithm is the next step of this process. When all the output of all the matrix’s column has been determined, the MSE will be calculated. If the MSE is still bigger than MSE target or the maximum iteration hasn’t been achieved yet, then the process should be continued. Else, the process is stopped, and the classification process is achieved.

In this study, the number of input nodes of the input layer and the hidden nodes of the hidden layer are 40 nodes and 4 nodes. Due to this reason, there are 164 adapted weight connecting the input layer to the hidden layer. Several trials have been done to this study using various learning rates. The best accuracy percentage on classifying the images based on its category is achieved by using $\alpha = 0.6049$ at the 235th iteration and $\alpha = 0.4719$ at the 134th iteration, resulting 100% of training and validation process on classifying the images.

| Table 1. The table of input nodes, hidden nodes, and adapted weights used in the study |
|-----------------------------------------------|
| **Input Nodes** | **Hidden Nodes** | **Adapted Weights** |
| 40 | 4 | 164 |

| Table 2. The percent accuracy results of image classification using PSN and LM Learning Algorithm |
|-----------------------------------------------|
| **Experiment Number** | **Learning Rate** | **n-th Iteration** | **Training Success (%)** | **Validation Success (%)** | **Mean Squared Error (MSE)** |
| 1 | 0.9562 | 172 | 100 | 70 | $9.8928 \times 10^{-6}$ |
| 2 | 0.3721 | 615 | 100 | 80 | $9.9987 \times 10^{-6}$ |
| 3 | 0.1824 | 129 | 100 | 80 | $9.8553 \times 10^{-6}$ |
| 4 | 0.8518 | 339 | 100 | 70 | $9.9364 \times 10^{-6}$ |
| 5 | 0.5866 | 961 | 100 | 80 | $9.9959 \times 10^{-6}$ |
| 6 | 0.7402 | 616 | 100 | 80 | $9.9996 \times 10^{-6}$ |
| 7 | 0.6049 | 253 | 100 | 100 | $9.8987 \times 10^{-6}$ |
| 8 | 0.4719 | 134 | 100 | 100 | $9.9839 \times 10^{-6}$ |
| 9 | 0.2534 | 191 | 100 | 75 | $9.8917 \times 10^{-6}$ |
| 10 | 0.0781 | 3550 | 100 | 75 | $9.9931 \times 10^{-6}$ |
Figure 4. MSE graphics for (a) $\alpha = 0.9562$  (b) $\alpha = 0.3721$

4. Conclusions

The focus of this study is to classify the retinal diseases, in this case, Age-related Macular Degeneration (AMD), Retinal Detachment (RD), and Diabetic Retinopathy (DR). The stages that have to be done to classify them are, image processing and reduction, and classification process. The results show 100% of training and validation success, using $\alpha = 0.6$ and $\alpha = 0.4$, in classifying retinal diseases using Levenberg-Marquardt Learning Algorithm for Pi Sigma Network (PSN) and Principal Components Analysis (PCA) methods.

References
[1] Sahu K K, Panigrahi S, and Behera H S 2013 A Novel Chemical Reaction Optimization Algorithm for Higher Order Neural Network Training Journal of Theoretical and Applied Information Technology
[2] Guler M and Sahin E 1994 A new Higher-Order Binary-Input Neural Unit: Learning and Generalizing Effectively via using Minimal Number of Monomials Proceedings of the Third Turkish Symposium on Artificial Intelligence and Neural Networks Proceedings. Middle East Technical University, Ankara, Turkey: pp. 51-60
[3] Shin Y and Ghosh J 1991 The Pi-sigma Network : An Efficient Higher-Order Neural Network for Pattern Classification and Function Approximation International Joint Conference on Neural Networks
[4] Sarabakha A, Imanberdiyev N, Kayacan E, Khanesar M A, and Hagras H 2017 Novel Levenberg-Marquardt Based Learning Algorithm for Unmanned Aerial Vehicles Information Sciences
[5] Santo R E 2012 Principal Component Analysis applied to digital image compression Einstein (SÀ£o Paulo 10(2):135-9
[6] Omar M, Khelifi F and Tahir M A 2016 Detection and classification of retinal fundus images exudates using region based multiscale LBP texture approach 2016 International Conference on Control, Decision and Information Technologies (CoDIT)
[7] Harini R and Sheela N 2017 Feature extraction and classification of retinal images for automated detection of Diabetic Retinopathy 2016 Second International Conference on Cognitive Computing and Information Processing (CCIP)
[8] Basuki A 2005 Pengolahan Citra Digital Menggunakan Visual Basic Graha Ilmu: Yogyakarta
[9] Damayanti A and Werdiningsih I 2018 Classification of tumor based on magnetic resonance (MR) brain images using wavelet energy feature and neuro-fuzzy model Journal of Physics: Conference Series, Volume 974, conference 1

[10] Aldrich C 2002 Exploratory Analysis of Metallurgical Process Data with Neural Networks and Related Methods Elsevier Science, British Library Cataloguing in Publician Data, Netherlands, Vol.1, pp. 56-57