PAPER • OPEN ACCESS

Potential of Heart Disease Detection Based on Iridology

To cite this article: C Yohannes *et al* 2020 *IOP Conf. Ser.: Mater. Sci. Eng.* **875** 012034

View the article online for updates and enhancements.

Potential of Heart Disease Detection Based on Iridology

C Yohannes[*1], I Nurtanio[1], and K C Halim[1]

[1]Department of Informatics, Faculty of Engineering, Universitas Hasanuddin, Makassar Indonesia

[*]Email: christoforus@unhas.ac.id

Abstract. Every year the number of deaths caused by heart disease continues to increase. This is mostly caused by misdiagnosis or misinterpretation of the heart disease symptoms. Therefore, people need to be aware of this disease by maintaining a healthy lifestyle and conducting regular checks on the potential for heart disease. This early examination can be done using the iridology method, namely by analyzing the iris of the eye. This paper presents the implementation of computer vision and machine learning of iridology to detect the potential for heart disease. This system uses Canny edge detection and Principal Component Analysis (PCA) to extract features in the iris region of the eye, and Backpropagation Algorithm of Artificial Neural Networks to create the predictive model. There are 110 data used in this system, consisting of 55 eye images from subjects suffering from heart disease and 55 images of normal subjects. The data is divided into 88 data for training and 22 data for testing. The proposed system produced accuracy up to 95.45% for the test data using the sigma 0.3 for canny edge detection, 50 principal components, 50 hidden neurons, and 0.01 for the error limits.

1. Introduction

Iridology is a method that can show health problems in various organs in the body by analyzing the colored areas in the eye, or also called iris. Iridology was popularized by Dr. Ignatz Van Peczely in 1881 in his book "Discoveries in the Field of Natural Science and Medicine". This method has been practiced for hundreds of years in various forms, including by utilizing digital imaging to determine the patient's health condition [1].

Cardiovascular and heart disease has become the main cause of about 17 million deaths in the world each year. WHO on its publication "The Atlas of Heart Disease and Stroke" stated that by 2020, heart disease and stroke will become the leading cause of both death and disability worldwide. They also recommend several precautions against the risk of this disease, one of which is by conducting periodic health checks [2]. Therefor this paper introduces a system for early detection of potential heart disease based on iridology.

This paper presents a system that uses Canny edge detection and Principal Component Analysis (PCA) as a method for extracting cardiac features in the iris. Canny edge detection is used to detect features in the form of blemishes in the heart region of the iris, while PCA is used to extract the strongest features found in the Canny. The results of the PCA are then used as input to the backpropagation neural network.

Research related to the detection of potential heart disease based on iridology has previously been conducted. The research uses the Canny edge detection and PCA to extract features. While the method used for classification is Support Vector Machine. This research achieved 100% accuracy for the training process, and 80% for the testing process [3].

Other research using artificial neural networks has also been carried out for cases of identification of autonomic nerve damage based on iridology. This research uses PCA for feature extraction and backpropagation neural network to do the classification. The data used in this study amounted to 80, consisting of 40 data from subjects suffering nerve damage and 40 data from normal subjects. This research produced the highest disease recognition rate of 83.33% [4].

2. Research Method

2.1. Iridology

Iridology is a method of medical examination popularized by Dr. Ignatz Van Peczely in 1881. The method of iridology works by analyzing the colored parts of the eye or commonly called Iris. Iris is the most complex network that can be seen from outside of the body. Iris is connected to hundreds of thousands of nerve endings, microscopic blood vessels, muscles, and other tissues. In other words, the iris is connected to all organs and tissues of the body through the brain and nervous system so that through the iris can be known the health conditions of various organs in the body.

Based on the iridology chart developed by Dr. Bernard Jensen, the heart organ can be clearly identified in the left eye with a position in the direction of 2 to 3 o'clock. Figure 1 shows the mapping of organs based on iridology chart by Dr. Bernard Jensen.
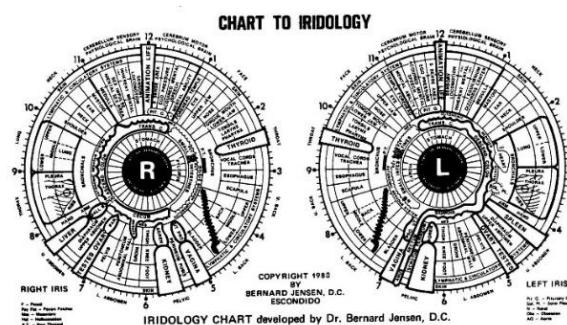

Figure 1. Iridology Chart by Dr. Bernard Jensen [5]

2.2. Dataset

In this study, 110 eye images consisting of 55 normal images and 55 images with heart disease were used. Data was obtained from one clinic that provides iridology service and was taken from a variety of different people in the same lighting conditions. The image used is in the .JPG format with a size of 500×500 pixels.

The first step in data preparation is to do manual isolation of the iris portion of the eye image for all existing images. After that, the pupil region is masked into black to facilitate the Region of Interest (ROI) selection process.

In this study, the data is divided into train and test data. For the training data, 88 data were used, consisting of 44 normal data and 44 data with heart disease. As for the test data, 22 were used consisting of 11 normal data and 11 data with heart disease.

2.3. Image Pre-processing

The initial stage in this system is data preprocessing. Preprocessing conducted in this study included converting RGB image into grayscale, histogram equalizing, smoothing, and pupil masking. Converting from RGB to grayscale is used to reduce the system computing load from three color channels to one channel. Histogram equalization method used in this study is Contrast Limited Adaptive Histogram Equalization (CLAHE). Histogram equalization is used to increase the contrast between the iris background and the iris fiber features. While smoothing is used to reduce noise arising from increased contrast in the previous stage. Smoothing method used in this study is a simple median filtering method. Figure 2 shows the results of the preprocessing performed.
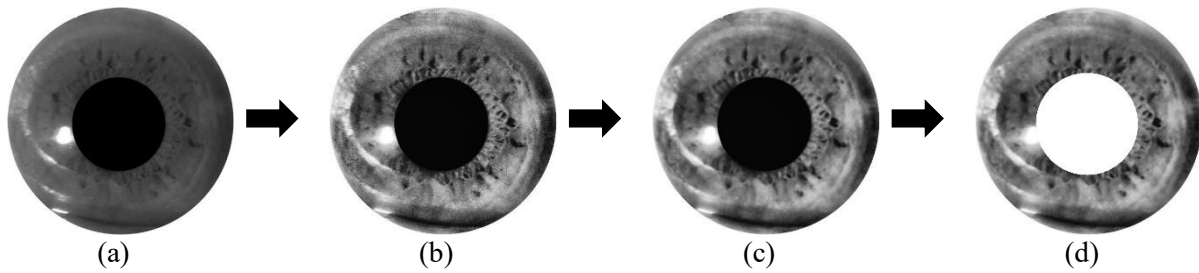
Figure 2. (a) Grayscale Image, (b) Histogram Equalized Image, (c) Smoothened Image, (d) Masked Pupil Image

2.4. Region of Interest (ROI) Selection

ROI is an area that contains features that will be used to recognize the condition of the heart in the iris. Based on the information of iridologists, the ROI position of the heart in the iris image is in the direction of 2 to 3 o'clock, so the system will only take the region as input. The ROI selection process is carried out automatically by the system.

The initial stage of choosing ROI is the process of transforming iris images after preprocessing into a linear image. The transformation used in this process is based on warpPolar function on [6]. This process will change the initial image of a circular iris into a linear form as shown in Figure 3(a). After that, the ROI of the heart can then be taken from a linear image. For the cardiac ROI, the portion of iris to be taken is only 50% of the iris band starting from the end of the pupil to the middle of the iris band as in Figure 3(b). The results of the ROI selection are then resized to a size of 50 × 50 pixels. The results of the ROI selection can be seen in Figure 3(c).
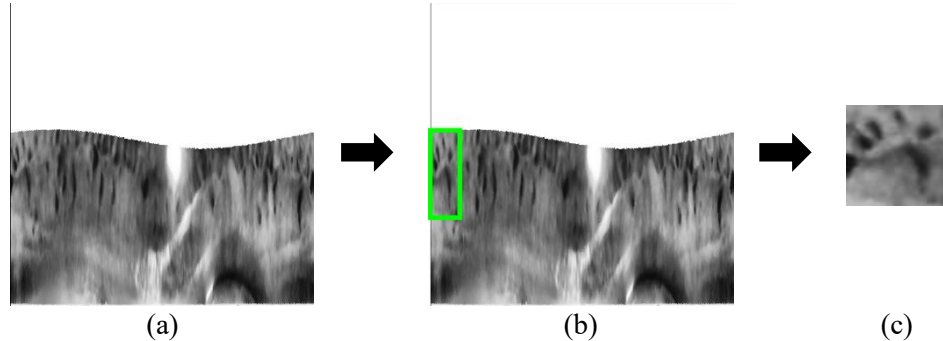
Figure 3. (a) Transformed Image, (b) Target ROI, (c) Resized ROI

2.5. Canny Edge Detection and Principal Component Analysis (PCA) for Feature Extraction

Canny edge detection is used to extract edges from iris fibers in ROI. The Canny edge detection method used has two main parameters namely *low threshold* and *high threshold* where the determination of these parameters is done automatically using a sigma value. Calculation of low threshold and high threshold values is done with the following Equation (1) based on [7]:

$$
\left.\begin{array}{l}
low = \max(0, (1.0 - sigma) \times median \\
high = \min(255, (1.0 + sigma) \times median
\end{array}\right\} \tag{1}
$$

In this study an experiment was conducted on five sigma Canny values, namely 0.1, 0.2, 0.3, 0.4, and 0.5. For each sigma value, different accuracy will be obtained in the system prediction results. The best accuracy is obtained at sigma value 0.3. The results of Canny edge detection are shown in the following Figure 4.

Figure 4. Canny Image for Sigma 0.3

All previous Canny edge detection images are then flattened to produce image array with a size of 1 × 2500 pixels, then combined into a dataframe. PCA is then used to retrieve the most important features or also called Principal Components (PC) contained in the data frame. The PC determination is done by determining the eigen value and eigenvector of each feature column. Some features with the highest eigen value can be taken as a PC to be used as input in neural networks. This PC selection process can also reduce the computational burden because it only takes a portion of all the features available. Broadly speaking, the PCA process is explained as follows according to [8].

The first stage in PCA is to standardize the range of values of data variables so that each variable has the same contribution to the analysis. The standardization process is done by subtracting the value of the variable with the average value of the variable as in Equation (2).

$$z = value - mean \qquad (2)$$

The next step is to calculate the covariance matrix to find out whether there is a relationship between variables in the data frame. This is also done to find out whether there is information redundancy in the dataframe so that it can be removed. This calculation is done for each dimension contained in the dataframe as shown in the Equation (3).

$$C(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)} \qquad (3)$$

Next is to calculate the eigen vector and eigen value of the covariance matrix to determine the PC. Determination of the features that become PCs is done by sorting the eigen vector based on the eigen value starting from the largest to the smallest. The greater the eigen value, the more information contained in the component.

After sorting the eigen vector, we can then choose the number of PCs that will be used as input in the neural network. This can be done by using the entire eigen vector or by removing a portion of the eigen vector that has a small eigen value. Discarding the eigen vector with a small eigen value can reduce the burden and system computing time because the amount of data to be processed is getting smaller.

After determining the eigen vector to be used, the feature reconstruction is carried out by multiplying the transposed standardized data by transposed feature vector as in Equation (4).

$$Final\ Set = FeatVector^T \times StandarizedOriDataset^T \qquad (4)$$

In this study, several experiments were carried out for the number of PCs including 10, 20, 30, 40, and 50 where the best accuracy was obtained on the 50 PCs.

2.6. Artificial Neural Network Backpropagation for Classification

Backpropagation algorithm is one algorithm that is widely applied to solve machine learning problems. This algorithm has an architecture consisting of several layers including the input layer, hidden layer, and output layer. Where hidden layer can consist of one or more hidden layers and each hidden layer can consist of more than two neurons. The architecture of backpropagation artificial neural networks is shown in Figure 5.
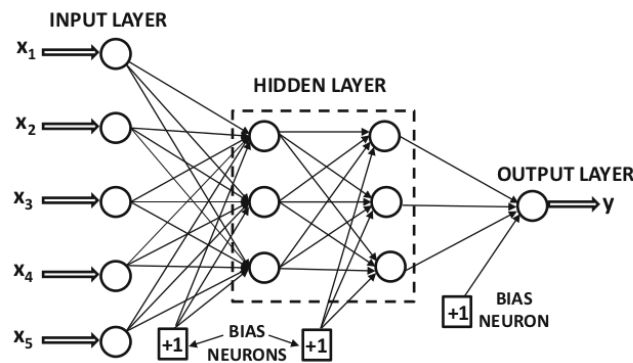
Figure 5. ANN Backpropagation Architecture

Backpropagation algorithm is divided into two stages, namely the training phase and the testing phase. The training phase aims to obtain the right weight values to be used in calculations so that they can produce accurate predictions. The training phase is divided into two processes, namely *feed forward* and *backpropagation*, where the feed forward process will be tasked to produce an output value, while backpropagation is to improve the initial weight value based on the error value obtained during the feed forward process. For the testing phase, the process that occurs is only the feed forward process that takes place using the final weight value obtained at the training stage [9].

The backpropagation algorithm in this study is used as a machine learning model that will produce predictions from iris images. The resulting prediction will be the presence or absence of potential heart problems experienced by the patient. In this case, the system output will be coded into 1 and 0, respectively representing potential and normal.

The architecture used in this study consists of an input layer, a hidden layer, and an output layer, where the number of neurons in the input layer follows the number of PCs used, and the number of neurons in the output layer is two for each class. As for the number of neurons in the hidden layer, an experiment is performed on several values including 10, 20, 30, 40, and 50. In addition to these parameters, this model also limits the number of iterations by 20,000 epochs, and uses error limits of 0.1 and 0.01. In this study, the best accuracy was obtained in model with the number of hidden neurons of 40 and 50 and at an error limit of 0.01.

3. Result and Analysis

The system was tested on machine with Intel(R) Core(TM) i7-5500U CPU @ 2.4GHz, 8GB of RAM, and operating on Windows 10 64-bit. The backpropagation model uses the same initial weight value for each experiment.

3.1. Results comparison for each Canny's sigma value

The best result from system testing are obtained at sigma value of 0.3 for the number of PCs as much as 50, 50 hidden neurons, and the error limit of 0.01 with an accuracy score of 95.455%. The accuracy comparison table between sigma values is shown in Table 1.

Table 1. Results comparison for each Canny's sigma value

Sigma Value	Classification Accuracy
0.1	72.727%
0.2	77.273%
0.3	95.455%
0.4	77.273%
0.5	77.273%

3.2. Results comparison for each Principal Component size

Determination of PC size greatly affects the performance of the backpropagation network model created. This is because the size of the PC also acts as the number of neurons contained in the input layer, so that it can affect the learning and testing of the system. The best accuracy of the system is obtained at the number of PC 50 for sigma Canny 0.3, 50 hidden neurons, and error limit 0.01. A comparison table of results for each PC size is shown in the following Table 2.

Table 2. Results comparison for each PC size

Principal Components	Classification Accuracy
10	68.182%
20	77.273%
30	59.091%
40	86.364%
50	95.455%

3.3. Results comparison for each number of hidden neuron

The number of hidden neurons can also affect the performance and results of the backpropagation network. The number of hidden neurons will affect the number of weights that connect between neurons at each layer so that it can affect the calculation results.

Table 3. Results comparison for each number of *hidden neurons*

Number of Hidden Neurons	Classification Accuracy
10	77.273%
20	81.818%
30	77.273%
40	95.455%
50	95.455%

In this study, the best accuracy result was obtained on the number of hidden neurons as much as 40 and 50 for sigma Canny 0.3, PC size 50, and error limit 0.01. The comparison table of the accuracy results for each number of hidden neurons is shown in Table 3.

3.4. Results comparison for each maximum error limit value
The maximum error limit value is used to limit the learning process carried out during training. Determination of error limits is done to prevent overfit so that the model can make better predictions against data that has never been encountered before. In this study, an experiment was conducted on two error limit values, 0.1 and 0.01, where the best accuracy was obtained at the error limit 0.01 with a sigma value of 0.3, PC size 50, and the number of hidden neurons 50. Accuracy comparison tables for each error limit value can be seen in Table 4.

Table 4. Results comparison for each maximum error limit value

Error Limit Value	Classification Accuracy
0.1	90.909%
0.01	95.455%

4. Conclusion
Detection of potential heart disease based on iridology can be done using Canny edge detection and Principal Component Analysis (PCA) for feature extraction, and Backpropagation Artificial Neural Networks as classifiers. The data used in this paper amounted to 110 data which were divided into 88 training data and 22 test data. The best accuracy was obtained in experiment using sigma 0.3, PC size of 50, hidden neurons by 50, and error limits 0.01, where the resulting accuracy was equal to 95.45%. Accuracy achieved successfully passed the accuracy of previous studies related to the detection of potential heart disease based on iridology.

Acknowledgment

References
[1]   "10 Conditions That Iridology Can Detect | Better Living Wellness Clinic," 20 Februari 2017. [Online]. Available: https://www.betterliving.co/2017/02/20/10-conditions-that-iridology-can-detect/.

[2]   "WHO publishes definitive atlas on global heart disease and stroke epidemic," World Health Organization, 23 September 2004. [Online]. Available: https://www.who.int/mediacentre/news/releases/2004/pr68/en/.

[3]   L. I. Permatasari, A. Novianty and T. W. Purboyo, "Deteksi Gangguan Organ Jantung Menggunakan Komputerisasi Iridologi Dengan Metode Klasifikasi SVM (Detection of Heart Organ Disorders Using Computerized Iridology With SVM Classification Method)," 2016.

[4]   E. S. Wicaksono, I. Santoso, A. A. Zahra and R. R. Isnanto, "Identifikasi Kerusakan Saraf Autonomik Melalui Citra Iris Mata Menggunakan Ekstraksi Ciri Analisis Komponen Utama (PCA) dan Jaringan Saraf Tiruan Perambatan Balik (Identification of Autonomic Nerve Damage through Iris Eye Images Using Feature Extraction Analysis of Main Components (PCA) and Back Propagation Neural Networks)," 2017.

[5] B. Jensen, Iridology Simplified: An Introduction to the Science of Iridology and Its Relation to Nutrition, Book Pub Co, 2011.

[6] "OpenCV Geometric Image Transformations," [Online]. Available: https://docs.opencv.org/3.4/da/d54/group__imgproc__transform.html#ga49481ab24fda a0ffa4d3e63d14c0d5e4.

[7] A. Rosebrock, "Zero-parameter, automatic Canny edge detection with Python and OpenCV," 6 April 2015. [Online]. Available: https://www.pyimagesearch.com/2015/04/06/zero-parameter-automatic-canny-edge-detection-with-python-and-opencv/.

[8] L. I Smith, A Tutorial on Principal Components Analysis, 2002.

[9] C. C. Aggarwal, Neural Networks and Deep Learning: A Textbook, Springer International Publishing, 2018.