International Journal of Mechanical, Industrial and Aerospace Sciences
ISSN: 2517-9950
Vol:6, No:9, 2012

Straight Line Defect Detection with Feed Forward Neural Network

S. Liangwongsan, A. Oonsivilai

***Abstract*—**Nowadays, hard disk is one of the most popular storage components. In hard disk industry, the hard disk drive must pass various complex processes and tested systems. In each step, there are some failures. To reduce waste from these failures, we must find the root cause of those failures. Conventionall data analysis method is not effective enough to analyze the large capacity of data. In this paper, we proposed the Hough method for straight line detection that helps to detect straight line defect patterns that occurs in hard disk drive. The proposed method will help to increase more speed and accuracy in failure analysis.

***Keywords*—**Hough Transform; Failure Analysis; Media; Hard Disk Drive

I. Introduction

Hard Disk Drives (HDD) are components for data storage used in computers. They are response to various uses that the demand of HDD is inverse variation to the cost.

Therefore, the HDD industry has to balance demand with high quality production, as such to increase capacity and production technology. The defined process of HDD production is complex. It consists of various components, namely read - head, media for data storage, and motor. Due to its complexity, there are some failures occurring in each process. Most commonly found failure is from defect on the media which cannot be repaired and wastes production cost. Failure Analysis (FA) is utilized to identify the failure of defect on media and classify the pattern of defect. FA helps technicians to access problems rapidly, decreasing waste and production cost. However, FA where failures are analyzed by human is not rapid enough. Consequently, this study is to introduce a method of Failure Pattern Detection based on the Hough Transform and Image Processing for detecting straight line pattern on media to help detecting defect in HDD media for the loaded case of FA process.

II. Hough Transform

The Hough transform is a technique used to find shapes in a binary digital image. By Hough Transform it is possible to find all kind of shapes that can be mathematical expressed, for instance lines, circles and ellipses, but only straight lines will be considered here. If having a white pixel in a binary image, infinity many straight lines can go through that single pixel, and each of these lines can go through other white pixels in the same image, and the more white pixels on the same line the more is this line represented in the image.

S .Liangwongsan is with the School of Machanical Engineering, Suranaree University of Technology, Nakhon Ratchasima Thailand (e-mail: sirisak.rong@gmail.com).

A. Oonsivilai is with Center of Excellence in Electric Energy, Smart Materials, and Health Science. Postharvest Technology Research Center, School of Electrical Engineering, Institute of Engineering, Suranaree University of Technology, Nakhon Ratchasima, 30000 Thailand. (e-mail: anant@sut.ac.th).

This is the principle of the Hough transform for straight lines. As mentioned above a shape can be found if a mathematical expression can be set for the shape, and in this case where the shape is a straight line, an expression can be set as:

$$y = a * x + b \tag{1}$$

Where a is the slope, and b is where the line intersects the y-axis. These parameters, a and b, can be used to represent a straight line as single point (a, b) in the parameter-space spanned by the two parameters a and b. The problem by represent a line as a point in the (a, b) parameter-space, is that both a and b goes toward infinity when the line becomes more and more vertical, and thereby the parameter space becomes infinity large. Therefore it is desirable to find another expression of the line with some parameters that have limited boundaries. It is done by using an angle and a distance as parameters, instead of a slope and an intersection.

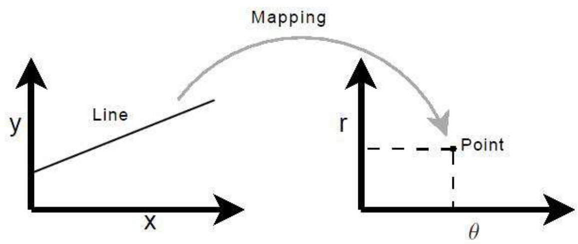
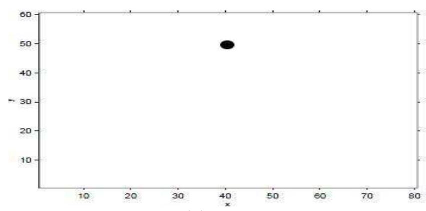
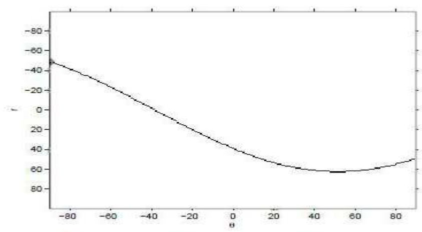


Fig. 1 Mapping of one unique line to the Hough space

An important concept for the Hough transform is the mapping of single points. The idea is that a point is mapped to all lines that can pass through that point. This yields a sine-like line in the Hough space. The principle is illustrated for a point p0 = (40, 30) in Figure3.



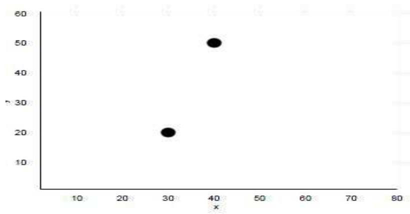
(a) Point p0



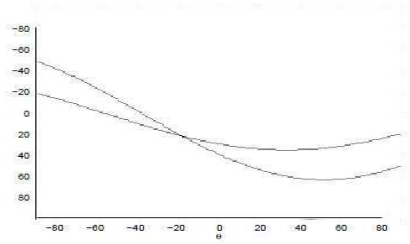
(b) p0 represented in the Hough space

International Journal of Mechanical, Industrial and Aerospace Sciences
ISSN: 2517-9950
Vol:6, No:9, 2012

Fig. 2 Transformation of a single point (p0) to a line in the Hough space. The Hough space line represents all possible lines through p0.



(a) Points p0 and p1



(b) p0 and/or p1 represented in the Hough space

Fig. 3 Transformation of two points (p0 and p1) to two lines in the Hough space. The intersection of the Hough space lines indicates the line that pass through both p0 and p1.

If the distance ρ (*rho*) is the distance from the origin to the line along a vector perpendicular to the line, and the angle θ (*theta*) is the angle between the x-axis and the r vector (see Figure 1), Equation 1 can be written as:

$$y = -\frac{\cos(\theta)}{\sin(\theta)} * x + \frac{\rho}{\sin(\theta)} \tag{2}$$

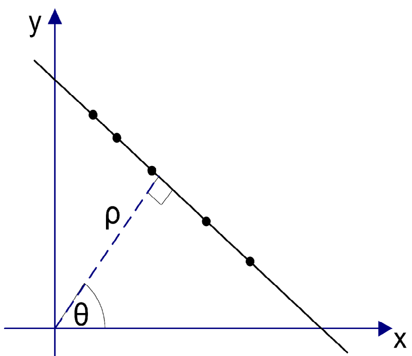


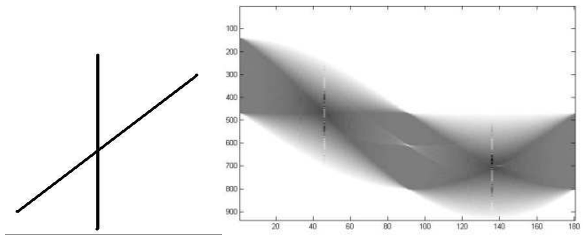
Fig. 3 *Rho* and *theta* representation of a straight line. Each line has a unique parameter set (ρ , θ)

The expressions here, instead of *a* and *b*, is found by trigonometrically calculations. To get an expression of ρ , Equation 2 can be rearranged to:

$$\rho = x * \cos(\theta) + y * \sin(\theta) \tag{3}$$

Contrary to when the parameters is *a* and *b*, the values that ρ and θ can have are limited to: θ ∈ [0,180] in degrees or θ ∈ 2 [0,π] in radians, and ρ ∈ 2 [−*D*,*D*] where *D* is the diagonal of the image. A line can then be transformed into a single point in the parameter space with the parameters θ and ρ, this is also called the Hough space.

If, instead of a line, having a pixel in an image with the position (x, y), infinity many lines can go through that single pixel. By using Equation 3 all these lines can be transformed into the Hough space, which gives a sinusoidal curve that is unique for that pixel. Doing the same for another pixel, gives another curve that intersect the first curve in one point, in the Hough space. This point represents the line, in the image space, that goes through both pixels. This can be repeated for all the pixels on the edges, in a edge detected image.



a) b)

Fig. 4 An example of straight line (a) and Hough space (b)

III. Propose Method

A. Preprocessing

Raw data of media defect are created and stored in database in each local hard disk factory. We used those data for input of image and preprocess image by initially cropped to size 100x100 resolutions and used smoothing image, edge detection, filter noise, by Canny Edge Detection Algorithm.

The Canny algorithm uses an optimal edge detector based on a set of criteria which include finding the most edges by minimizing the error rate, marking edges as closely as possible to the actual edges to maximize localization, and marking edges only once when a single edge exists for minimal response [20]. According to Canny, the optimal filter that meets all three criteria above can be efficiently approximated using the first derivative of a Gaussian function.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4}$$

$$\frac{\partial G(x,y)}{\partial x} \alpha x e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \frac{\partial G(x,y)}{\partial y} \alpha y e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{5}$$

where: *g(k,l)* = convolutional kernel
I(x,y) = original image
I'(x,y) = filtered image
2N + 1 = size of convolutional kernel

Both the Gaussian mask and its derivative are separable, allowing the 2-D convolution operation to be simplified. This optimization is not limited to software implementation only, but applies to hardware implementation as well, as shown in the next section.

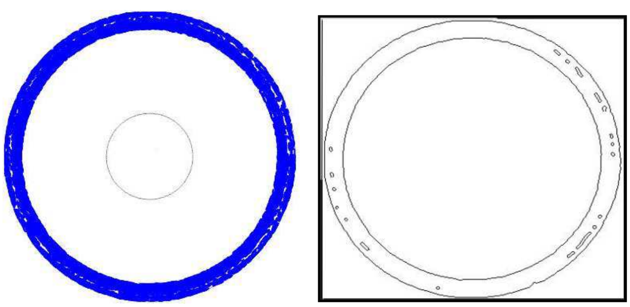
The non-maximal suppression stage finds the local maxima in the direction of the gradient, and suppresses all others, minimizing false edges. The local maxima is found by comparing the pixel with its neighbors along the direction of the gradient. This helps to maintain the single pixel thin edges before the final thresholding stage.

Instead of using a single static threshold value for the entire image, the Canny algorithm introduced hysteresis thresholding, which has some adaptivity to the local content of the image. There are two threshold levels, t_h, high and t_l, low where $t_h > t_l$. Pixel values above the t_h value are immediately classified as edges. By tracing the edge contour, neighboring pixels with gradient magnitude values less than t_h can still be marked as edges as long as they are above t_l. This process alleviates problems associated with edge discontinuities by identifying strong edges, and preserving the relevant weak edges, in addition to maintaining some level of noise suppression. While the results are desirable, the hysteresis stage slows the overall algorithm down considerably.

The performance of the Canny algorithm depends heavily on the adjustable parameters, σ, which is the standard deviation for the Gaussian filter, and the threshold values, t_h and t_l. σ also controls the size of the Gaussian filter. The bigger the value for σ, the larger the size of the Gaussian filter becomes. This implies more blurring, necessary for noisy images, as well as detecting larger edges. As expected, however, the larger the scale of the Gaussian, the less accurate is the localization of the edge. Smaller values of σ imply a smaller Gaussian filter which limits the amount of blurring, maintaining finer edges in the image. The user can tailor the algorithm by adjusting these parameters to adapt to different environments with different noise levels.

Results can be further improved by performing edge detection at multiple resolutions using multi-scale representations, similar to the Marr-Hildreth algorithm [2]. This is achieved using different standard deviations, which correspond to different resolution versions of the image. Edges have zero crossing a multiple scale values. Combining maxima information from different scales allows better classification of true edges.

Convolution at multiple resolutions with large Gaussian filters require even more computation power. This may prove to be challenging to implement as a software solution for real-time applications.



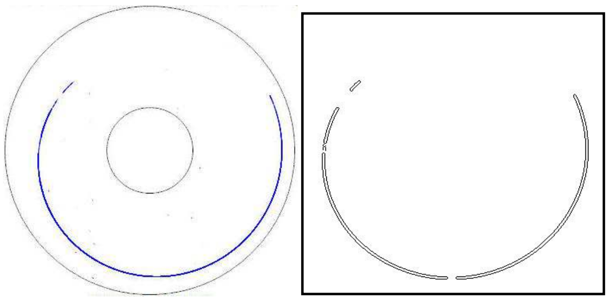


Fig. 5 defect circle pattern after passed Canny Edge Detection Algorithm

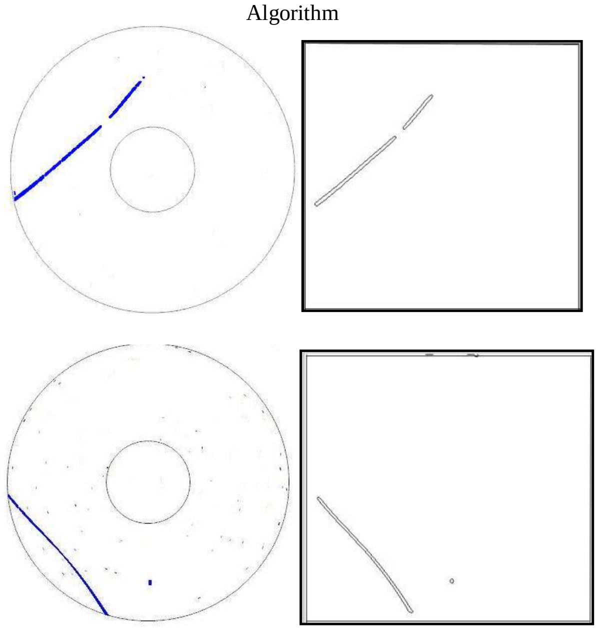


Fig. 6 defect straight line pattern after passed Canny Edge Detection Algorithm

B. Feature extraction by Circle Hough transform

This step used Hough transform techniques to identify position of arbitrary shape. The purpose of this technique is to find imperfect instances of object in parameter space within a certain class of shape by voting procedure. This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform. For each input image in the x-y plane, a spherical filter is applied and the outcome of this is a set of points in the parameter space. These points represent the centers of probabilistic circles of pre-defined radii in the x-y plane. Mapping is done for each point in the original space to a corresponding point in the original space to a corresponding point in accumulator space and increment the value of the accumulator bin.

To determine the areas where most Hough space lines intersect, an accumulator covering the Hough space is used. When an edge point is transformed, bins in the accumulator is incremented for all lines that could pass through that point. The resolution of the accumulator determines the precision with which lines can be detected.

In general, the number of dimensions of the accumulator corresponds to the number of unknown parameters in the Hough transform problem. Thus, for ellipses a 5-dimensional space is required (the coordinates of its center, the length of its major and minor axis, and its angle). For lines 2 dimensions suffice (r and θ). This is why it is possible to visualize the content of the accumulator.

C. Classification by Feed Forward Neural Network (FFNN)

The Back propagation neural network used here is Feed Forward Neural Network (FFNN). FFNN was the first and arguably simplest type of artificial neural network devised. In this network, The information moves in only one direction, forward, from the input notes, though the hidden nodes (if any) and do the output nodes. The number of input Layers, hidden layers and output layers are ad adjusted to fit the data point to the curve, During the training phase the training data in the accumulator array is fed into the input layer. The data is propagated to the hidden layer and ten to the output layer. This is the forward pass of the back propagation algorithm. Each node in the hidden layer gets input from all the nodes from input layer which are multiplexed with appropriate weights and summed. The output of the hidden node is the nonlinear transformation of the resulting sum. Similar procedure is carried out in the output layer. The output values are compared with target values are compared with target values and the error between two is propagated back towards the hidden layer. This is the backward pass of the back propagation algorithm. The procedure is repeated to get the desired accuracy. During the testing phase the test vector is fed into the input layer. The output of FFNN is compared with training phase output to match the correct one. This can serve as need to do recognition of facial objects.

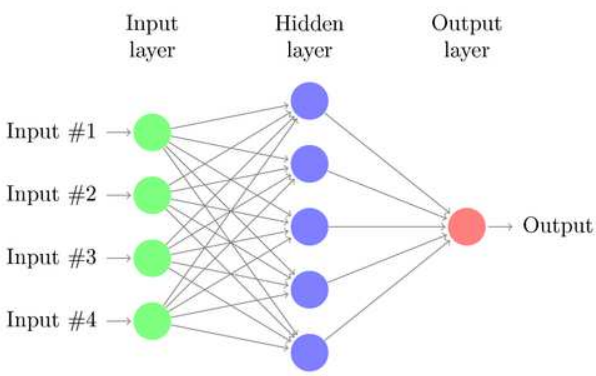


Fig. 7 Block diagram feed forward neural network

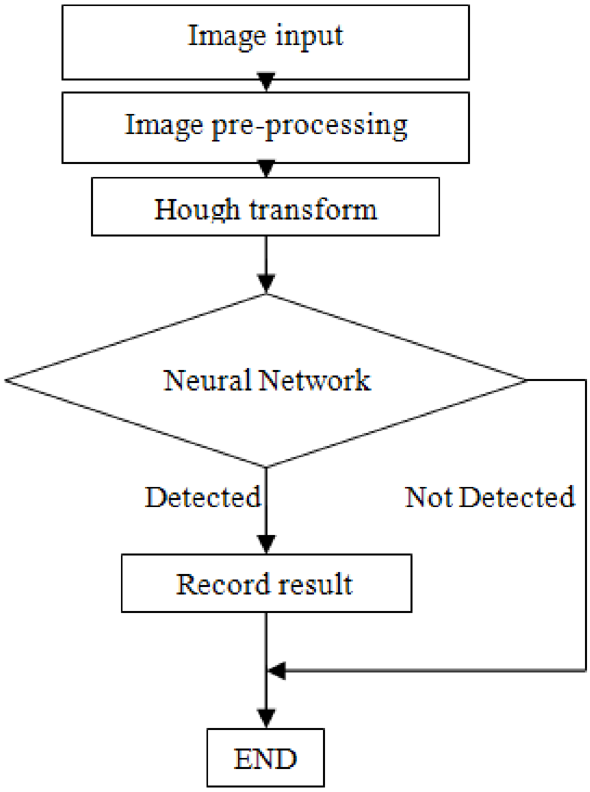


Fig. 8 Block diagram of the entire process

IV. Experiment Result

The 140 sample pictures of media defect with straight line pattern were sampled and pre – processed by image processing techniques. The samples were smoothed by Canny edge detection technique, and their noise were eliminated by Gaussian filter. It was shown that the Canny edge detection was able to both eliminating the noise and smoothing the border edge. To decrease the sample borders, the magnitude and orientation of them were calculated by the first order differentiation, then the nonmaxima suppression was employed in searching for the magnitude of gradient. To identify the pixel border and connect the border, double thresholding algorithm was employed. The sample images then were resized to 100x100 pixels for a fast image processing response.

The peak values of Straight line detection algorithm were searched in the Accumulator arrays. A threshold value was defined to represent the boundary of the straight line pattern. Then, the maximum values from the accumulator array were compared with the threshold to identify for the straight line pattern. Fig. 9 demonstrated the straight line pattern defect and the accumulator. Fig. 10 demonstrated the circle pattern depicting the peak value voted from the accumulator array, that the defect was the straight line pattern. A feed forward neural network with 30 hidden layers was trained from Table I with hidden layer is 30 network is best accuracy from this neural network the accuracy obtained is very high with computed value of 99.29% and the performance plot of neural network each layer as shown in figure 11.

International Journal of Mechanical, Industrial and Aerospace Sciences
ISSN: 2517-9950
Vol:6, No:9, 2012

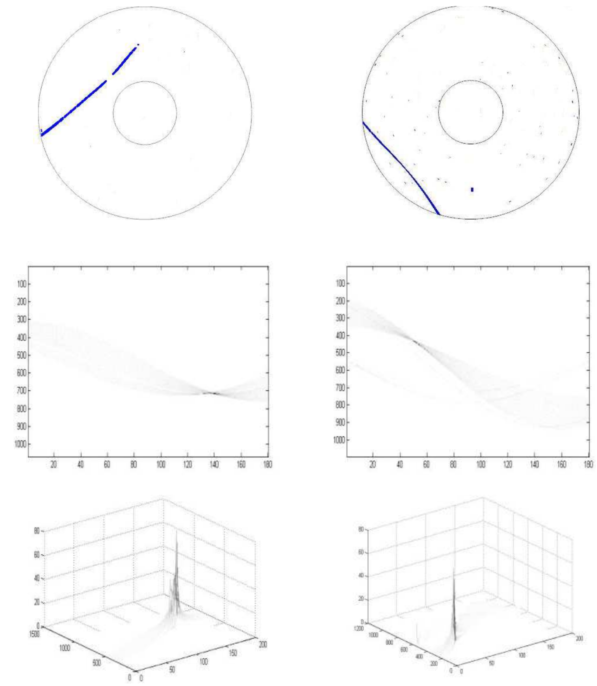


Fig. 9 Straight line pattern & Accumulator 2D,3D view

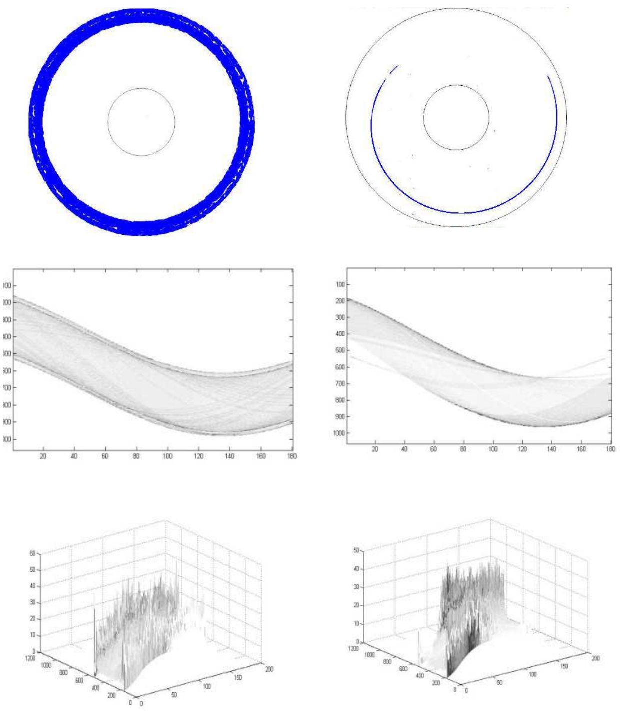


Fig. 10 Sample of circum & radial pattern & Accumulator 2D,3 Dview (this group is not straight line algorithm not detect)

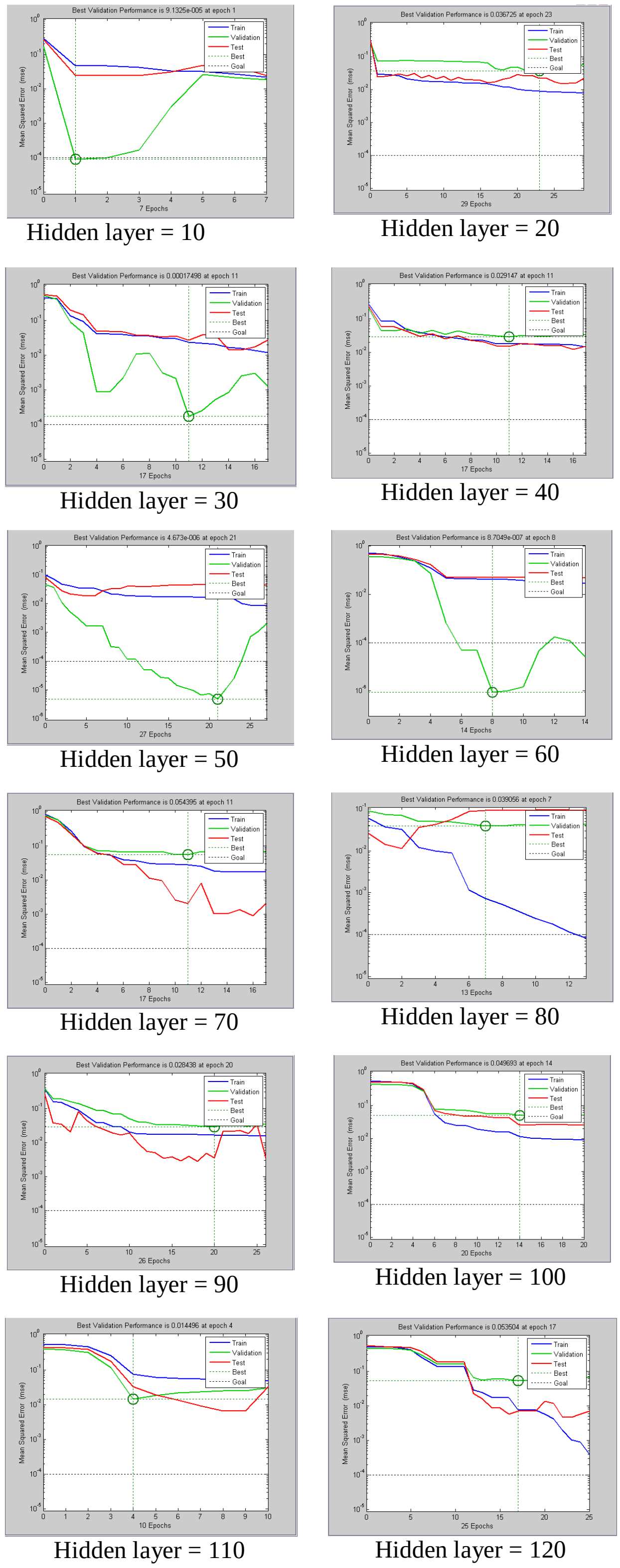


Fig. 11 Performance plot of neural network

TABLE I
COMPARE AUUCURACY 10-120 HIDDEN LAYER

NO.OF IMAGE	NO. HIDDEN LAYER	NO. OF DETECTED	FAULTY	%ACCURACY
140	10	133	7	95.00%
140	20	137	3	97.86%
140	30	139	1	99.29%
140	40	138	2	98.57%
140	50	138	2	98.57%
140	60	133	7	95.00%
140	70	134	6	95.71%
140	80	138	2	98.57%
140	90	138	2	98.57%
140	100	135	5	96.43%
140	110	135	5	96.43%
140	120	134	6	95.71%

V.CONCLUSION

This study's aim was to introduce an algorithm for straight line pattern detection on HDD media based on the Hough Transform method and feed forward neural network (FFNN). The experiment demonstrated a satisfying result. The developed algorithm was advantage for the failure analysis of HDD production process, such as help to access problems rapidly, decrease waste, and reduce the production cost of HDD industry. To minimize the search time duration and yield the effective search, the border lines of samples were identified by the pre – process of the image processing feed forward neural network (FFNN) method.

ACKNOWLEDGMENT

This research was supported by research and development institute funds, Suranaree University of Technology, Nakhon Ratchasima, Thailand.

REFERENCES

[1] Jack Bresenham. A linear algorithm for incremental digital display of circular arcs. Communications of the ACM February, Volume 20,Number 2, February 1977.
[2] Chester F. Carlson. Lecture 10: Hough circle transform. Rochester Institute of Technology: Lecture Notes, October 11, 2005.
[3] Rafael C. Gonzalez and Richard E. Woods. Digital Image Processing. Prentice Hall, 2007. ISBN 0-13-168728-x.
[4] Carolyn Kimme, Dana Ballard, and Jack Sklansky. Finding circles by an array of accumulators. Communication of the ACM, Volume 18, Number 2, February 1975.
[5] Bryan S. Morse. Lecture 15: Segmentation (edge based, hough transform). Brigham Young University: Lecture Notes, 2000.
[6] Shylaja S S,K N Balasubramanya Murthy, S Natarajan Nischith,Muthuraj R,Ajay S.Feed Forward Neuarl Network Base Eye Localization and Recognition Using Hough Transform. IJACSA Vol. 2,No.3,pp104-109, March 2011
[7] Ballard DH, "Generalizing the Hough transform to detect arbitrary shapes", In the proceedings of Pattern Recognition, pp. 111–122, 1981.
[8] J. Illingworth and J. Kittler. "A survey of the Hough Transform", In the proceedings of International Conference on Computer Vision, Graphics and Image Processing, pp. 87–116, 1988.
[9] JR Bergen and H Shvaytser, "A probabilistic algorithm for computing Hough Transforms", In the Journal of Algorithms, pp.639–656, 1991.
[10] N Kiryati, Y Eldar, and AM Bruckstein, "A probabilistic Hough Transform", In the proceedings of Pattern Recognition, pp. 303–316,1991.
[11] Duda RO, Hart PE, "Use of the Hough Transform To Detect Lines and Curves In Pictures", In the Communications of ACM, pp. 11–15, 1972.
[12] Kimme, C., D. Ballard and J. Sklansky, "Finding Circles by an Array of Accumulators", In the Communications of ACM, pp. 120–122, 1975.
[13] Mohamed Roushdy "Detecting Coins with Different Radii based on Hough Transform in Noisy and Deformed Image" In the proceedings of GVIP Journal, Volume 7, Issue 1, April, 2007.
[14] Cheng Z. and Lin Y., "Efficient technique for ellipse detection using restricted randomized Hough transform", In the proceedings of the International Conference on Information Technology (ITCC'04), Vol.2, pp.714-718, 2004.
[15] Yu Tong, Hui Wang, Daoying Pi and Qili Zhang, "Fast Algorithm of Hough Transform- Based Approaches for Fingerprint Matching", In the Proceedings of the Sixth World Congress on Intelligent Control and Automation, Vol. 2, pp.10425- 10429, 2006.
[16] Lam, W.C.Y., Yuen, S.Y., "Efficient technique for circle detection using hypothesis filtering and Hough transform", IEE Proceedings of Vision, Image and Signal Processing, vol. 143, Oct. 1996.
[17] Torii, A. and Imiya A., "The Randomized Hough Transform Based Method for Great Circle Detection on Sphere", In the Proceedings of Pattern Recognition, pp. 1186-1192, 2007.
[18] Cheng Z. and Lin Y., "Efficient technique for ellipse detection using restricted randomizedHough transform", Proc. of the International Conference on Information Technology(ITCC'04), Vol.2, pp.714-718, 2004.
[19] Yip, R.K.K.; Leung, D.N.K. and Harrold,S.O., "Line segment patterns Hough transform for circles detection using a 2-dimensional array", International Conference on Industrial Electronics, Control, and Instrumentation, Vol.3, pp. 1361-1365, 1993.
[20] Canny, J., "A Computational Approach to Edge Detection", IEEE Transactions on Pattern Analysis and Machine Intelligence,Vol.8, No. 6, Nov. 1986.
[21] Simon Just Kjeldgaard Pedersen , Circular Hough Transform, Aalborg University, Vision, Graphics, and Interactive Systems, November 2007.
[22] Maar, D., Hildreth E., "Theory of edge detection",*Proceedings Royal Soc. London*, vol. 207, 187-217,1980
[23] Sirisak L.,Boonruang M.,Ratchadaporn O.,Anant O. "Straight Line Defect Detection Algorithm with Hough Transform" WORLD ACADEMY OF SCIENCE,ENGINEERING AND TECHNOGY ISSUE 60,DECEMBER 2011
[24] Sirisak L.,Boonruang M.,Ratchadaporn O.,Anant O. "Extracted Circle Hough Transform and Circle Defect Detection Algorithm" WORLD ACADEMY OF SCIENCE,ENGINEERING AND TECHNOGY ISSUE 60,DECEMBER 2011

Sirisak Liangwongsan, was born in Saraburi Province,Thailand,in 1982 He received B.Eng from Rajamangara University of Technology Thanyaburi., Thailand, in Telecommunication Engineering. And currently a student M. Eng. in school of Mechanical Suranaree University of Technology, Thailand.

Anant Oonsivilai, was born in Khon Kaen Province, Thailand, in 1963. He received his B.Eng from Khon Khan University, and M.Eng. from King Mongkut Institute of Technology North Bangkok, Thailand and PhD. From Dalhousie University, Canada, all in Electrical Engineering, in 1986,1992 and 2000, respectively. Dr. Oonsivilai is currently an Assistanst Professor in school of Electrical Engineering, Suranaree University of Technology, Thailand.