Finding Optimal Transformation Function for Image Thresholding Using Genetic Programming

Shaho Shahbazpanahi, Shahryar Rahnamayan, IEEE Senior Member

***Abstract*—In this paper, Genetic Programming (GP) is employed to obtain an optimum transformation function for bi-level image thresholding. The GP utilizes a user-prepared gold sample to learn from. A magnificent feature of this method is that it does not require neither a prior knowledge about the modality of the image nor a large training set to learn from. The performance of the proposed approach has been examined on 147 X-ray lung images. The transformed images are thresholded using Otsu's method and the results are highly promising. It performs successfully on 99% of the tested images. The proposed method can be utilized for other image processing tasks, such as, image enhancement or segmentation.**

Keywords*—*Genetic Programming, Otsu Thresholding, Optimum, Transformation function

I. Introduction

SINCE human antiquity, image and pictorial information have been a part of human life. The first humans used images to communicate. Nowadays with the presence of computers, intelligence from an image has become part of the fabric human existence. Generally speaking, image processing techniques can be divided in two main categories. In the first category, the output of image processing techniques is an image which is evaluated or used by a user. Image de-blurring or noise filtering are examples form this category which human uses of the processed image. In the second category, the output image of an image processing algorithm is fed to another algorithm to process. For example, the output of an image enhancement algorithm can be input for an image segmentation method. The proposed method in the current creates an optimal transformation function which is suitable for bi-level image thresholding.

Machine learning methods are classified in two major categories: *supervised* and *unsupervised* methods. In supervised learning, a desired output is provided. The algorithm learns from provided input and their corresponding output in training phase and generate the inferred function, which can be applied for mapping untrained new samples. An optimal scenario will allow for the algorithm to accurately specify the class labels for unseen instances in testing phase. On the contrary, in unsupervised methods the desired outputs are not provided. According to this classification, since gold sample is provided by an expert user, our proposed approach is categorized in the supervised learning class [1], [2], [3].

There is no a general agreement clearly defined as an edge between image processing and other areas such as image analysis or computer vision [4]. However, Gonzalez believes that a differentiation can be achieved by characterizing input and output of image processing tasks [4]. He has suggested that the chain of image processing can be classified in the three levels as follows: low-, medium, and high- levels. A low-level image processing is considered as preliminary techniques. Image enhancement and image filtering have been considered in this level. The mid-level includes image segmentation, object representation, and object description. Image segmentation deals with dividing image into similar region. Object representation and description methods extract all the shape information of the original object, for computer processing. Finally, object recognition is a high-level processing.

Mahmood and Tariq proposed a method to restore a degraded image from its degradation version and to detect object in a video stream [5]. Genetic

S. Shahbazpanahi was graduated from the Department of Electrical and Computer Engineering, University of Ontario Institute of Technology, Oshawa, ON, Canada.
E-mail: shaho.shahbazpanahi@uoit.ca
S. Rahnamayan is faculty member in the Department of Electrical, Computer, and Software Engineering, University of Ontario Institute of Technology, Oshawa, ON, Canada.
E-mail: shahryar.rahnamayan@uoit.ca

programming (GP) is applied for this purpose which arithmetic operations are used as a function set and the feature vectors, which are small neighborhood around each pixel and constant sets are used as a terminal set. The performance of Mahmood's algorithm has been compared with Richarson-Lucy (RL) and Winner filter methods. The results indicate that the method is more accurate and robust than the RL and the Winner filter.

Wang et al. [6], [7] constructed morphological operations sequence to enhance infrared vein fingerprint image and to detect object in a binary image by means of linear GP. The basic morphological and logical operations are in function set and regular and irregular structuring elements are used as terminal set. The algorithm has been compared to the Quintana's algorithm with the respect to the correlation between obtained image and ground-truth image. The result highlights that the proposed algorithm has 12% more improvement compared to the Quintana's algorithm.

II. Background Review

Genetic Programming was proposed by John Koza [8] in 1990, which is a population based algorithm to generate computer programs. By simulating evolution by natural selection, GP produces generations of successively fitter populations [9]. GP's main steps are shown in Algorithm 1 [9].

Algorithm 1 . GP

1: Randomly create an initial population of individuals (structures).
2: **repeat**
3: Evaluate each individual and find fitness value.
4: Select two individuals as parents from the population by giving a higher chance to fitter parents to be selected.
5: Create new individuals by applying the genetic operations (crossover, mutation, cloning).
6: **until** stopping criteria is met.
return best-so-far individual

GP uses a syntax tree to define expressions. Each tree is constructed from some set of subtrees which are called *subordinates*. In a syntax tree, the variable and constants are its leaves, while the internal nodes are *functions* .

To initialize a population in a GP, mainly there are three methods [12]: *full, grow, and ramped half-and-half* methods.

In evolutionary algorithms, biological operators are applied to individuals which have better fitness values in the generation. The commonly used methods for the parent selection is tournament selection.

Genetic programming has three operations, namely, crossover, mutation, and reproduction; which can be applied, based on their probabilities to construct new trees.

III. Proposed Method

In the training phase, an optimum enhanced transform function is obtained by applying the GP on a X-ray of lung and corresponding gold image. Then, in testing phase, the transformation function is applied to some untrained new samples to verify the robustness of the proposed method.

As Fig. 1 indicates, the values of parameters set in the first step are tabulated in Sec. IV-A. In the next step, an initial population is randomly generated which contains individuals with a tree structure. Each tree is built by the set of terminals (input) and functions The first generation can use any method for initialization. However, to support a better initialization, the ramped-half-and-half is applied. By this way, each individual in the generation represents a transformation function.

Enhanced image is obtained by applying the transformation function to the input image. Then, the enhanced image is thresholded by the Otsu algorithm, [13] Then, the binary images are compared with the gold image and its fitness value is calculated.

Now individuals are selected based on their fitness values. Then, GP operations, which are crossover, mutation, and reproduction, are applied on selected individuals. After that, the fitness value of new individual are calculated.

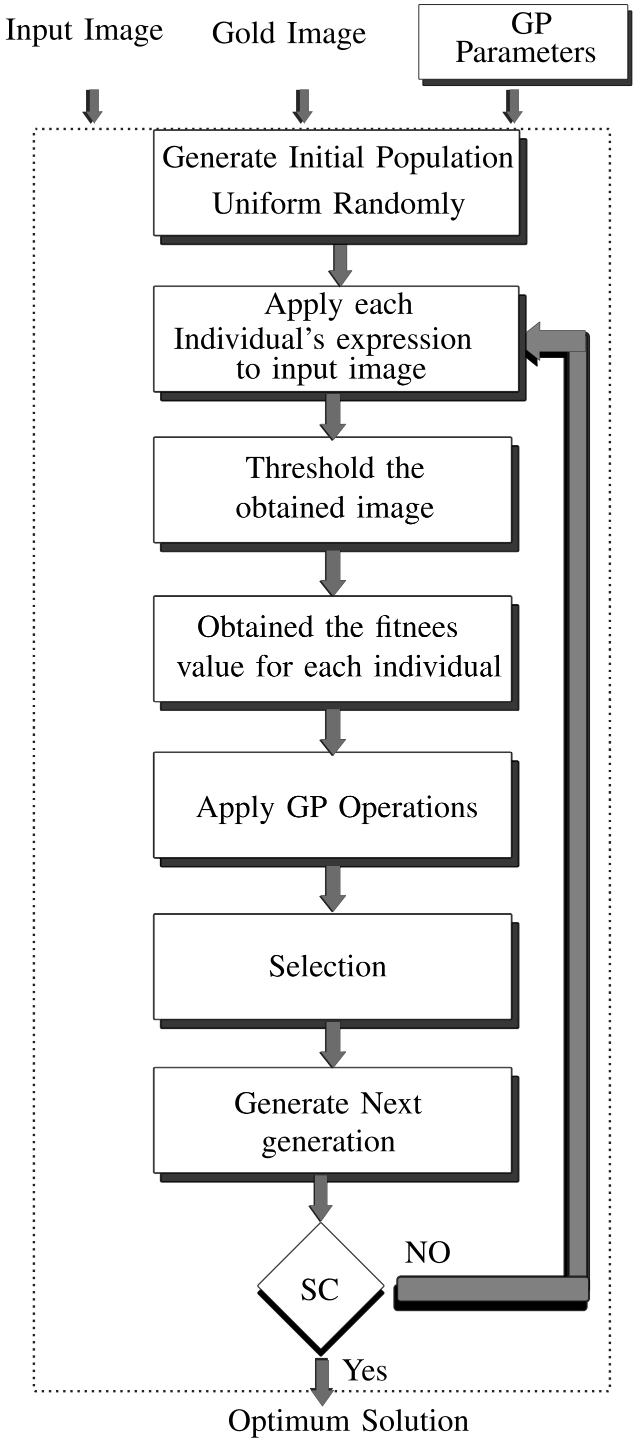


Fig. 1: Finding optimal transformation function using Genetic Programming. SC: Stopping Condition

In the next step, GP fills the new generation based on generated individual candidates. The steps continue until the stopping criteria is met.

IV. Experimental Results

In this section, the proposed algorithm is applied to X-ray lung images, the algorithm generates an optimum transformation function in training phase. Then, the obtained transformation function is applied to 147 untrained images.

A. *Parameter settings*

The parameters for proposed method are given in Table I. The stopping criteria is set to number of generation. The function set for GP has been set to basic functions, such as, subtraction, addition, and division. The GP uses ramped half-and-half method for initialization. The maximum tree size in GP is set to 512 nodes, this means trees whose number of nodes are bigger than 512 cannot be included in the next generation.

TABLE I: GP's parameter settings for image enhancement

Number of generation	100
population size	10
Function set	$uminus, max, min, *, \div, ln, log, sin, \cos, e^x$
Terminal set	constant numbers, input image
Initial method	Ramped Half-and-Half
Maximum size of tree	512 nodes

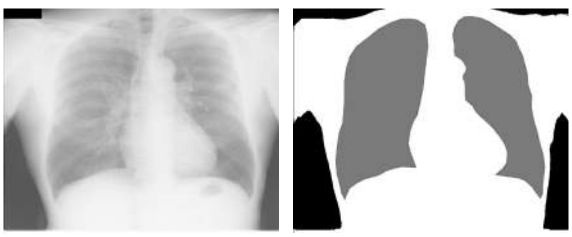
In order to evaluate the quality of individuals in population, we need to define an objective function to measure the similarity of each individual with the gold image. So, let the fitness value of each individual is computed as follows:

$$\max f = \sum_{i=1}^{N} \sum_{j=1}^{M} |I_{i,j} \cap G_{i,j}| + W \sum_{i=1}^{N} \sum_{j=1}^{M} |I_{i,j} \cap R_{i,j}|, \tag{1}$$

where $|I_{i,j} \cap G_{i,j}|$ obtains the number of pixels that have the same values in gold image G and thresholded image I; and $|I_{i,j} \cap R_{i,j}|$ calculates the number of pixels that have the same values in the region-of-interest (ROI) of gold image and thresholded image I, W is weight factor to emphasize the effect of correct detection of the ROI. The value of W is related to the size of the ROI in image, if the region is big this value can be small and vice versa. In our experiment, the W is set to 10. The number of common pixels between the gold image and the thresholded image should be maximized by the GP.

B. *Training phase*

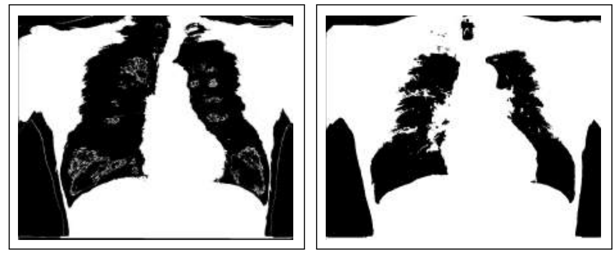
Training for image enhancement has been accomplished by applying GP on the gray scale input image and corresponding gold image, as shown in Fig. 2.



(a) (b)

Fig. 2: Training set of image enhancement. (a) Input image, (b) ground-truth (gold) image.

Fig. 4 demonstrates tree structure of the optimum transformation function (tree) of the proposed method. The tree uses functions as internal nodes and input the gray-level intensity of pixel x as leaves of tree. The input image is a gray scale image which its pixel are in the range $[0, 255]$. The obtained tree has intron nodes which are those nodes in the syntax tree with no contribution. In a tree structure, these introns are inevitable. For example, the most right branch of tree represents $\ln(\sin(-|-|\log_2(x)||))$. This branch has three introns node: two mines, $-$, operations and an absolute operation, $|.|$. Corresponding fitness plot and Results of optimization are shown in Fig. 5 and Fig. 3, respectively. The lung has been thresholded more accurate compared to the situation which no transformation was performed before applying the Otsu thresholding.



(a) (b)

Fig. 3: Training set result by applying learning image enhancement algorithm.
(a) Threshold result of Otsu method after image enhancement by generated optimal transformation function, (b) Thresholded of original image by Otsu method without applying transformation.

C. *Testing phase*

The optimal transformation function is applied to 147 X-ray sample images [14] and the results are thresholded by Otsu method. The thresholding results for four samples are shown in the Fig. 6. The first column is the sample test image and second column contains the corresponding ground-truth image. The third and forth columns are the result of threshold of Otsu thresholding without and with optimal transformation, respectively. The numerical results for all 147 sample images in the test set are presented in Table II which indicates that performance of the suggested algorithm is much better than stand alone Otsu method except in image number 61 that Otsu algorithm outperforms our method. This fitness results obtained by evaluating the proposed fitness function (i.e., similarity measure) between gold and corresponding output image.

D. *Verification test*

We can experientially check the (sub)optimality of the found transformation function (at least its local optimality), if one operation is changed, the output result should differ. The optimum transformation function is as follows:

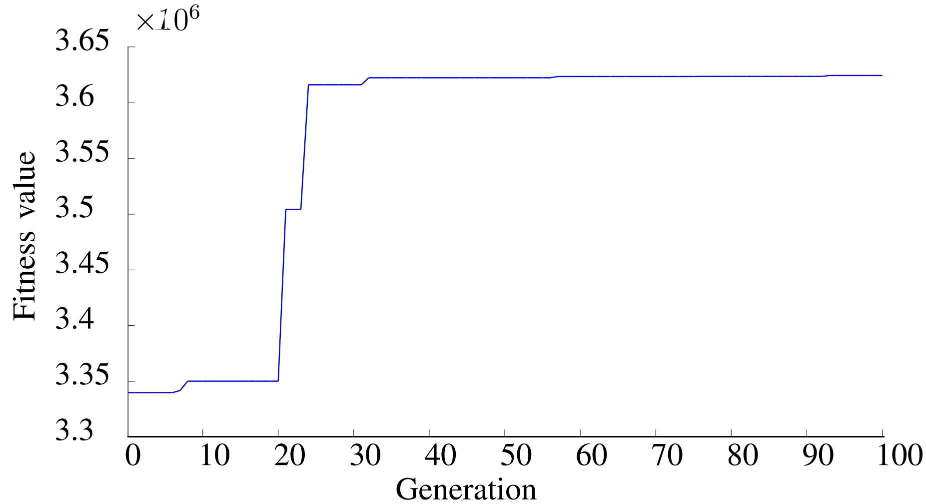


Fig. 5: Fitness plot.

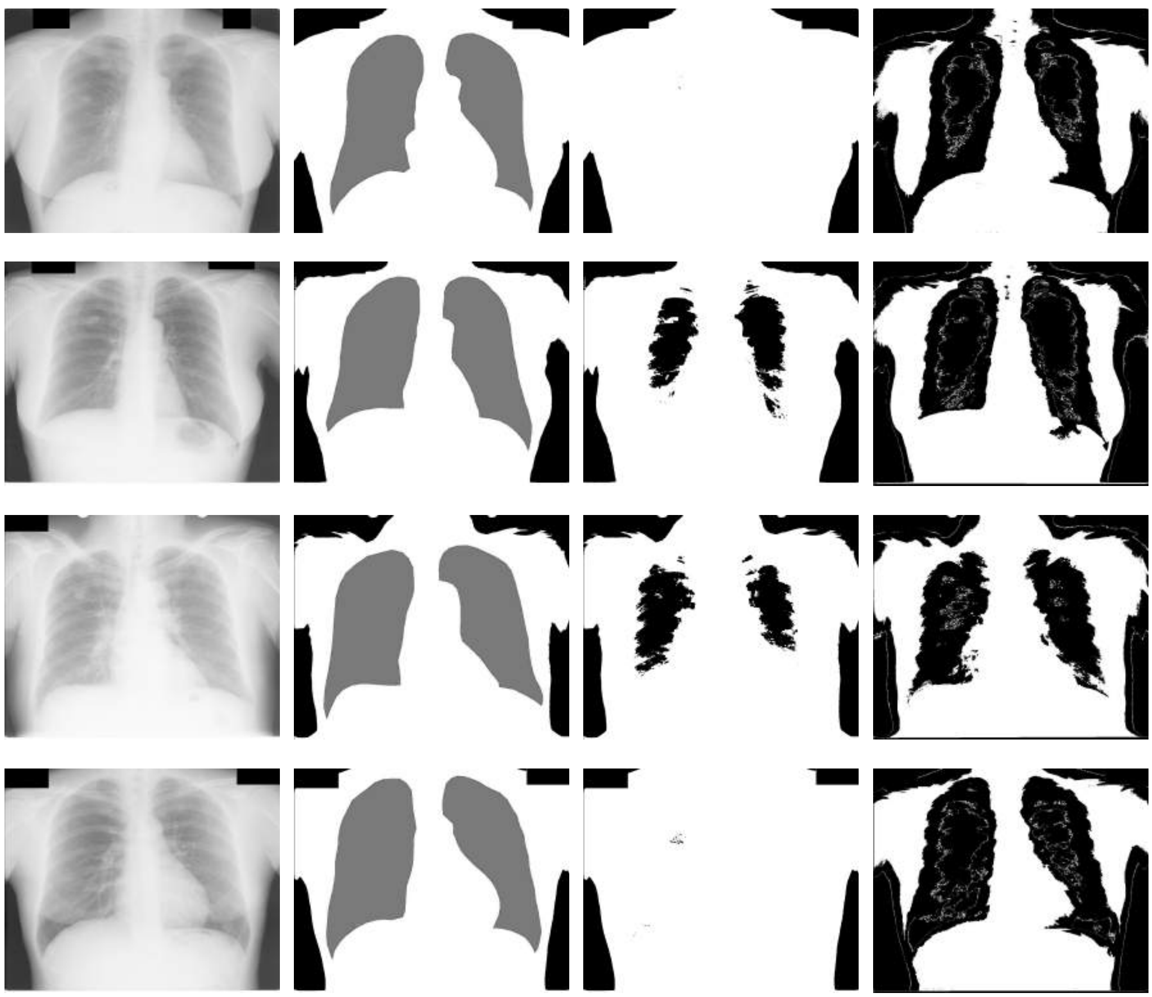


Fig. 6: Some sample results of X-ray lung Images. First column, left-to-right: Original image, ground-truth image, Thresholding image by Otsu without image transformation, thresholding by Otsu after optimal transformation.

TABLE II: Numerical results for Lung X-ray threshold test samples, the better result for each case is shown by a bold font.

Image No.	Stand alone Otsu (%)	Proposed Method (%)	Image No.	Stand alone Otsu (%)	Proposed Method (%)	Image No.	Stand alone Otsu (%)	Proposed Method (%)
1	62.9	**83.3**	50	21.0	**63.2**	99	65.4	**85.0**
2	78.4	**87.1**	51	52.0	**85.1**	100	88.1	**91.4**
3	80.4	**84.1**	52	20.7	**85.1**	101	80.9	**90.9**
4	20.3	**88.8**	53	79.6	**86.3**	102	15.0	**80.4**
5	83.5	**83.6**	54	41.9	**83.9**	103	76.6	**93.4**
6	48.8	**86.6**	55	22.3	**82.3**	104	61.4	**84.5**
7	59.1	**82.7**	56	75.3	**87.7**	105	26.2	**76.7**
8	14.0	**91.2**	57	29.2	**89.1**	106	80.9	**93.0**
9	55.2	**87.7**	58	80.3	**83.1**	107	89.4	**93.5**
10	28.7	**68.6**	59	80.5	**88.1**	108	51.8	**77.7**
11	89.1	**92.0**	60	61.2	**87.5**	109	77.6	**90.0**
12	78.0	**90.1**	61	**89.4**	88.8	110	17.9	**80.4**
13	22.0	**84.2**	62	19.2	**81.1**	111	13.1	**74.1**
14	48.2	**74.9**	63	77.7	**93.5**	112	72.8	**89.4**
15	21.1	**84.0**	64	71.0	**88.3**	113	16.3	**77.9**
16	86.3	**89.3**	65	79.6	**90.4**	114	26.8	**89.7**
17	71.7	**80.1**	66	53.8	**86.5**	115	74.4	**83.1**
18	89.7	**94.3**	67	79.0	**88.7**	116	45.0	**84.7**
19	17.9	**79.8**	68	17.9	**83.7**	117	51.6	**86.3**
20	18.7	**87.1**	69	29.3	**83.5**	118	79.9	**88.8**
21	80.9	**91.7**	70	15.7	**69.0**	119	30.2	**87.4**
22	19.1	**87.1**	71	25.4	**84.8**	120	87.0	**94.7**
23	29.2	**81.1**	72	83.9	**85.6**	121	74.5	**91.4**
24	21.9	**83.7**	73	78.5	**87.7**	122	17.8	**82.8**
25	52.1	**87.4**	74	91.0	**94.1**	123	52.8	**89.2**
26	78.4	**88.8**	75	15.5	**84.3**	124	42.3	**77.2**
27	42.0	**86.7**	76	17.6	**87.6**	125	20.9	**86.1**
28	72.0	**88.4**	77	70.7	**91.9**	126	76.4	**94.3**
29	75.1	**81.8**	78	71.1	**90.9**	127	23.2	**85.1**
30	47.5	**86.6**	79	29.0	**89.9**	128	24.2	**82.0**
31	19.2	**87.3**	80	19.5	**89.8**	129	84.0	**92.3**
32	78.1	**85.8**	81	62.9	**89.6**	130	18.3	**76.9**
33	54.6	**90.4**	82	61.2	**90.1**	131	53.8	**87.8**
34	77.0	**83.3**	83	51.7	**88.7**	132	66.1	**89.0**
35	83.4	**91.4**	84	22.9	**87.0**	133	63.4	**73.4**
36	87.4	**89.8**	85	71.4	**85.8**	134	49.2	**87.2**
37	24.7	**83.8**	86	69.5	**88.1**	135	76.5	**92.2**
38	80.7	**91.0**	87	76.0	**90.7**	136	25.4	**85.5**
39	25.2	**86.8**	88	60.1	**90.2**	137	32.0	**79.3**
40	71.6	**83.0**	89	75.6	**90.8**	138	21.2	**80.6**
41	65.8	**85.5**	90	18.7	**84.4**	139	15.8	**77.4**
42	74.6	**76.5**	91	62.7	**90.8**	140	81.1	**92.1**
43	21.3	**76.4**	92	84.1	**89.2**	141	17.7	**81.0**
44	21.3	**74.8**	93	63.3	**82.8**	142	18.7	**88.7**
45	79.6	**83.5**	94	75.7	**93.1**	143	82.2	**87.5**
46	18.9	**91.1**	95	42.5	**82.9**	144	58.0	**83.5**
47	65.4	**79.2**	96	18.5	**77.3**	145	55.0	**83.0**
48	37.9	**83.1**	97	18.2	**70.6**	146	60.9	**76.7**
49	75.3	**89.0**	98	49.7	**87.7**	147	65.9	**87.5**

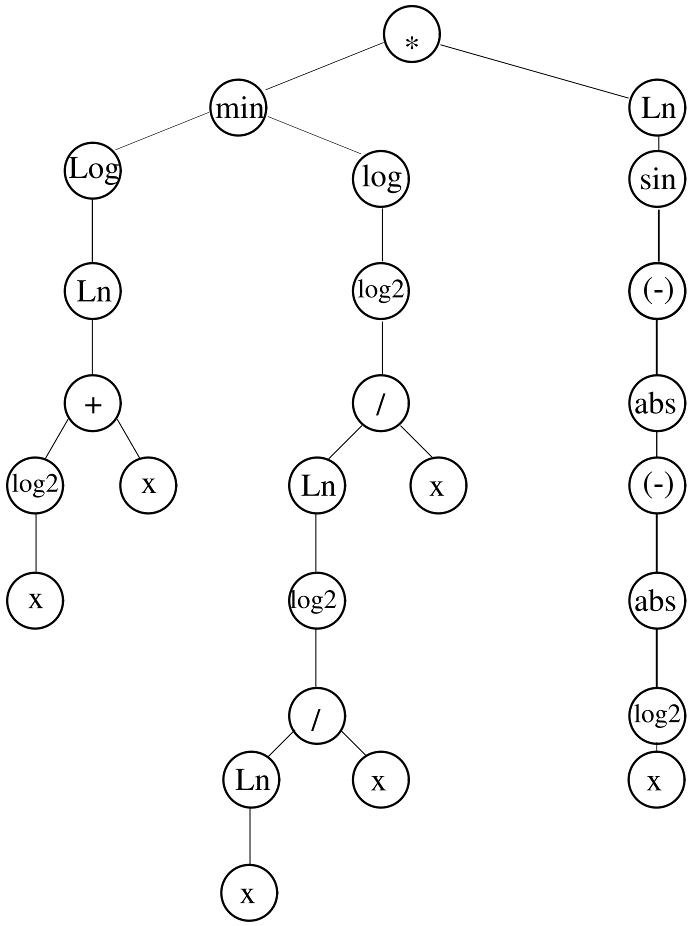


Fig. 4: The obtained optimum tree for X-ray image enhancement for image thresholding

$$
\begin{aligned}
f(x) = min[&\log(\ln(\log_2(x) + x)) \\
&, \log(\log_2(\frac{\ln(\log(\frac{\ln(x)}{x}))}{x}))] \\
&+ \ln(\sin(-|-|\log_2(x)||)).
\end{aligned} \tag{2}
$$

If the plus operator is replaced with mines operation in above equation, the following equation is resulted:

$$
\begin{aligned}
f(x) = min[&\log(\ln(\log_2(x) - x)) \\
&, \log(\log_2(\frac{\ln(\log(\frac{\ln(x)}{x}))}{x}))] \\
&+ \ln(\sin(-|-|\log_2(x)||)).
\end{aligned} \tag{3}
$$

Fig.7.c is result which is degraded when we apply a minor change in the obtained equation.

V. Conclusion

The fitness results obtained by evaluating the proposed fitness function (i.e., similarity measure) between gold and corresponding output image. The results show a general improvement of 38% improvement in detecting lung by applying the transformation function compared to results of stand alone Otsu.

The main part of the proposed method is to find an optimum transformation function that by applying this function to X-ray lung images, the lungs in thresholded image, are extracted more accurately. A comprehensive test set with 147 images has been performed to analyze the robustness and performance of the proposed approach. For 146 out of 147 images (99% of images), the lungs are segmented better, compared to the results of standalone Otsu method. The future work will be extension of proposed methods to enhance other image processing techniques set such as image filtering and image segmentation.

References

[1] Hinton, Geoffrey E and Sejnowski, Terrence Joseph, *Unsupervised learning: foundations of neural computation*. MIT press, 1999.

[2] Reed, Russell D and Marks, Robert J, *Neural smithing: supervised learning in feedforward artificial neural networks*. Mit Press, 1998.

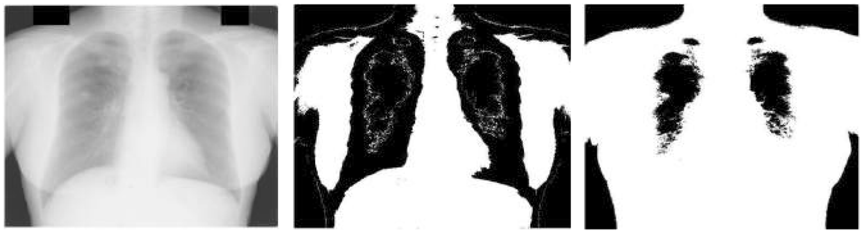
[3] Rahnamayan, S and Tizhoosh, HR and Salama, MMA, *Robust object segmentation using genetic optimization of morphological processing chains*. Proceedings of the 5th WSEAS international conference on Signal, speech and image processing, World Scientific and Engineering Academy and Society (WSEAS), pp. 248–253, 2005.

[4] Gonzalez, Rafael C and Richard, E, *Digital Image Processing*, 3rd Ed. Prentice Hall Press, ISBN 0-201-18075-8, 2002.

[5] Mahmood, M. Tariq and Majid, Abdul and Han, Jongwoo and Choi, Young Kyu, *Genetic programming based blind image deconvolution for surveillance systems*. Engineering Applications of Artificial Intelligence, Elsevier, pp. 1115–1123 2012.

[6] Wang, Jun and Tan, Ying, *A novel genetic programming algorithm for designing morphological image analysis method*. Advances in Swarm Intelligence, Springer, pp. 549–558, 2011.

[7] Wang, Jun and Tan, Ying, *Morphological image enhancement procedure design by using genetic programming*. Proceedings of the 13th annual conference on Genetic and evolutionary computation, pp. 1435–1442, ACM, 2011.



(a) (b) (c)

Fig. 7: The result of a sample verification test. (a) Input Image, (b) Thresholding result for applying equation 2 to input image, after applying the Otsu method, (c) Thresholding result for applying equation 3 to input image after applying the Otsu method.

[8] Koza, John R, *Genetic Programming: vol. 1, On the programming of computers by means of natural selection* . MIT press, 1992.

[9] Silva, Sara and Almeida, Jonas, *GPLAB-a genetic programming toolbox for MATLAB*. Proceedings of the Nordic MATLAB conference, pp. 273–278, 2003.

[10] Langdon, William B and Poli, Riccardo, *Foundations of genetic programming*. Springer, 2002.

[11] Poli, Riccardo and Langdon, William B and McPhee, Nicholas F and Koza, John R, *Genetic programming: An introductory tutorial and a survey of techniques and applications*, . University of Essex, UK, Tech. Rep. CES-475, 2007.

[12] Storn, Rainer and Price, Kenneth, *Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces*. Journal of global optimization, Springer, pp. 341–359 1997.

[13] Otsu, Nobuyuki, *A threshold selection method from gray-level histograms*. Automatica, 1975.

[14] Japanese Society of Radiological Technology, *Lung X-Ray Dataset.*, available at: : http://www.jsrt.or.jp/jsrt-db/eng.php.