Connected Component Labeling Algorithms for Gray-Scale Images and Evaluation of Performance using Digital Mammograms

Roshan Dharshana Yapa[†] **and Koichi Harada**[††],
Department of Information Engineering, Graduate School of Engineering,
Hiroshima University, Japan.

Summary
The main goal of this paper is to compare performance of connected component labeling algorithms on grayscale digital mammograms. This study was carried out as a part of a research for improving efficiency and accuracy of diagnosing breast cancer using digital mammograms. Three connected component labeling algorithms developed by Jung-Me Park [8], Kenji Suzuki [16] and Kesheng Wu [9], were used for this study. However, these algorithms had been tested and evaluated on binary images. Necessary modifications were introduced to those original algorithms to use them with grayscale images. We used MATLAB to implement these algorithms. Among these algorithms Kedheng Wu's algorithm with necessary modifications for grayscale images and using some optimization techniques in MATLAB such as vectorization and Pre-Memory allocation, showed a significant outstanding performance on digital grayscale mammograms. We used 30 digital mammograms selected from MIAS database for the evaluation.

Key words:
Connected Component, gray-scale, Mammogram.

1. Introduction

Mammography is the only universally accepted tool for early detection of breast cancer in women experiencing no symptoms; and to detect and diagnose breast disease in women experiencing symptoms such as a lump, pain or nipple discharge. Mammogram can detect changes in the breast up to two years before a patient or a physician detects them. Currently a large number of researches are being done on mammography imaging techniques to improve the early detection of breast cancer and to improve the accuracy (sensitivity and specificity) in distinguishing non-cancerous breast conditions from the breast cancers [14].

Many computer image processing and computer vision techniques are being used in digital mammography image processing. Among them, initial pre-processing techniques applied on mammograms plays a vital role in improving the accuracy of image analysis and later processing steps. Connected component labeling is a very important tool used in pre-processing stages as well as in image analysis and in post processing stages [4]. After recognizing connected components of an image, every set of connected pixels having same gray-level values are assigned the same unique region label. After recognizing connected regions in a mammogram, region based multi-scale image analysis would be used for identification of suspicious (malignant / benign) lesions or calcifications [10].

In many reported studies, connected component labeling algorithms are applied on binary mammogram images. They have used histogram based segmentation methods to convert a grayscale image into binary image using manual or adaptive threshold value selection methods [1, 2, 7, 13]. This process has two drawbacks; (i) Due to conversion of the image from grayscale to binary image, high information loss will occur. (ii) Developing fully automated computer aided diagnosis (CAD) algorithms would be more complex; as they searches for unknown patterns in mammograms and hence it will be difficult to determine the threshold value to segment those objects; or else those segmentation algorithms will be very complex.

These facts motivated us to evaluate the performance of existing connected component labeling algorithms directly on grayscale images without converting it into binary images. Also this paper emphasizes the need for developing faster connected component labeling algorithms to process grayscale images without converting into binary images.

Next section of this paper briefs some existing algorithms for connected component labeling. Pros and cons of these algorithms are also discussed within this section. Section three explains changes introduced to these algorithms, so that they can be used with grayscale images. Section four outlines the experimental conditions used and the section five discusses results drawn from the experiment. Finally, the section six discusses about conclusions of this experiment.

Manuscript received June 5, 2008
Manuscript revised June 20, 2008

2. Connected Component Algorithms for Binary Images

Various algorithms for connected component labeling have been published in the literature. According to Suzuki[16] those algorithms can be classified in to four classes as; (A) Methods with repeated forward and backward passes over data. (B) Methods with two passes over data. These algorithms maintain a separate vector or an array to store equivalence information of labels and use search algorithms to resolve equivalences. (C) Methods using hierarchical tree structure to represent data. (D) Methods using parallel algorithms. Citing [16] the first two classes (A and B) of connected component labeling algorithms are representatives and suitable for ordinary computer architecture. Rest of this section briefly explains these algorithms. As these algorithms are used for binary images, this section considers pixel value 0 as belongs to background and pixel value 1 as belongs to an object of interest in a binary mages.

2.1. Rosenfeld and Pfalts Algorithm

This was the very first algorithm developed for connected component labeling. The original algorithm was developed by Rosenfeld and Pfalts in 1966 [15] which belongs to class B according to Suziki's classification. This algorithm scans the image from left to right and top to bottom. This algorithm uses 4-neighbor forward raster scan mask as shown in figure 1 (a).

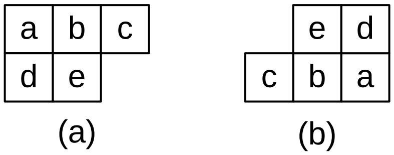

Fig 1: Typical masks of labeling of eight connected components. (a) Forward raster scan. (b) Backward raster scan.

Current pixel position of this scan mask is considered to be position "e". If the value of the current pixel position e is 0 then the mask is moved to the next scanning position. If current pixel e is 1 and all other 4-nbr positions are 0 then assign a new label to pixel "e". If two or more 4-nbr pixels are not zero then assign the minimum label of 4-nbrs to "e" and mark labels in 4-nbrs are as equivalent. A separate matrix is used to store equivalence details. After provisional labels have been identified, redundancies in labels are recognized by using an equivalence matrix (a binary matrix). Any undirected equivalence relations are satisfied three properties; namely, reflexivity, symmetry and transitivity. The equivalence matrix needs to be processed to resolve equivalence relations properties and remove label redundancies in label matrix. Then in the second pass each pixel is replaced by the final label assigned to its equivalence classes.

But one major drawback in this algorithm is the equivalence matrix can become unexpectedly large for large images and hence processing time may be prohibitively long [12, 8]. As a solution for this problem Park [8] has proposed an improvement for this algorithm by using divide and conquer technique.

2.2. Park's Algorithm

The main idea of this algorithm is to reduce the processing time greatly by reducing the size of the equivalence array used in [15]. This has been achieved by dividing the original image into NxN small regions and applying the Rosenfeld and Pfalts [15] algorithm to each region independently to generate local labels for each region. Then [Park et al]'s algorithm connects each region with its neighbor regions by resolving equivalences in region boundaries. This algorithm uses NxN pointer array Label_List[i] to point to arrays that maintain the *global labels* with respect to the entire image. Label_List[i] points to the array for Region[i] where each array element is a global label within the entire image and the index for each array element is a local label within Region[i]. According to Park [8], this algorithm performs faster than the Resenfeld and Pfalts [15] algorithm for binary images.

While studying this algorithm we found that Park [8] has overlooked to resolve one important boundary equivalence condition. Park [8] has discussed three conditions for resolve label equivalences on region boundaries when merging regions with its neighbor regions; (i) Merge of pixels values of neighboring regions with the first pixel of the current region. (ii) Merge of pixels in the first column of the current region and last column of the neighbor region on the west direction. (iii) Merge of the pixels in first row of the current region and the last row of the neighbor region on north direction of the current region. But according to this algorithm, an object similar to the figure 2 doesn't recognize as belongs to a same single region.

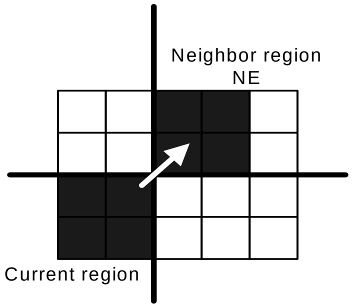

Fig 2: Merge of the last pixel in the first row of the current region and the last pixel in the first column of the neighbor region on the north-east side.

As a solution for this problem, it needs to consider possible equivalences of pixel values in last pixel of the first row of the current region and the last pixel of the first column of the neighbor region on the north-east as the forth condition for merging neighbor regions.

2.3. Conventional Forward and Backward Raster Scan Algorithm

This algorithm belongs to class A, according to Suzuki's [16] classification. Conventional algorithm [5, 6] in the class A repeats passes through a binary image *b(x,y)* in the forward and backward raster directions alternatively [17, 6]. Suppose that a binary image *b(x,y)* consists of pixel values F_o, indicating objects, and F_B, indicating the background; and that a provisional label *m* is initialized to one. First, following sequential local operations in the forward raster scan order, called the forward scan, are performed using the mask shown in the figure (1a) for eight-connected components.

$$g(x,y)=\begin{cases} F_B & \text{if } b(x,y)=F_B, \\ m,(m=m+1) & \text{if } \forall\{i,j\in M_S\}g(x-i,y-j)=F_B, \\ g_{\min}(x,y) & \text{otherwise,} \end{cases}$$

$$g_{\min}(x,y)=\min[\{g(x-i,y-j)\,|\,i,j\in M_S\}],$$

Where (*m=m+1*) indicates an increment of *m*, min(.) an operator calculating the minimum value, and M_S the region of the mask except the object pixel, i.e. *b(x-1,y-1), b(x,y-1), b(x+1,y-1), and b(x-1,y).*

Then the following operations in the backward raster scan order, called backward scan, are performed using the mask shown in Fig (1b).

$$g(x,y)=\begin{cases} F_B & \text{if } g(x,y)=F_B, \\ \min[\{g(x-i,y-j)\,|\,i,j\in M_S\}] & \text{otherwise,} \end{cases}$$

The forward and backward scans are repeated alternatively until no provisional labels change, and then the final labeled image can be obtained by assigning unique labels for each connected region.

One major problem associated with this algorithm is, labeling can be completed through plural scans in forward and backward directions. But, maximum number of scans required to complete the labeling has not been theoretically proven. However, the number of scans required depends on geometrical complexity of the image. Due to these reasons this algorithm may take unexpectedly long time to complete the labeling [16].

Suzuki [16] has proposed an improvement for this conventional algorithm by using an additional one dimensional table called *label connection table* to memorize label equivalences. This table is used successively during operations. This algorithm propagates provisional labels not only on the image but also on the label connection table. This reduces the number of required forward and backward scans necessary to complete the labeling by reflecting the connectivity of provisional labels at a geometrical distance, in the connection table.

In the first scanning step, Suzuki's [16] algorithm determines the provisional label for each pixel at position "*e*" (scan mask) as follows:

$$g(x,y)=\begin{cases} F_B & \text{if } b(x,y)=F_{B,} \\ m,(m=m+1) & \text{if } \forall\{i,j\in M_s\}g(x-i,y-j)=F_B, \\ T_{\min}(x,y) & \text{Otherwise} \end{cases}$$

$$T_{\min}(x,y)=\min[\{T[g(x-i,y-j)]\,|\,i,j\in M_S\}]$$

The label connection table is also updated simultaneously with the provisional label assignment as follows:

$$\begin{cases} \text{non-operation} & \text{if } b(x,y)=F_B, \\ T[m]=m & \text{if } \forall\{i,j\in M_s\}g(x-i,y-j)=F_B, \\ T[g(x-i,y-j)]=T_{\min}(x,y) & \text{if } g(x-i,y-i)\neq F_B. \end{cases}$$

Operations from the second scanning step are defined similarly with relevant modifications to the above formula.

Suzuki [16] has experimentally attested that, binary images complete labeling by no more than four scanning steps. This would improve the speed dramatically comparative to the conventional algorithm in class A. Furthermore Kesheng Wu [9] has proposed further improvement for the Suzuki's [16] algorithm by reducing the number of neighbors examined during the scanning steps and reducing the cost of union find algorithm by array based equivalence matrix rather than pointer based rooted trees.

2.4. Kesheng Wu's Algorithm

As the first strategy of this improvement, Kesheng Wu's [9] algorithm is based on the argument that, neighbor pixels of an image are correlated. Each of pixels in an image belongs to one of objects composed in the image and then one or many neighbor pixels of a current pixel would be related to each other. As proposed by Kesheng Wu [9], with appropriate supporting data structures, only one neighboring pixel is needed to determine the label of a new pixel. This can reduce the number of neighbors examined from four to one; but this is not the case for every pixel. However the average number of neighbor pixels scanned is usually less than four. Based on this strategy, Kesheng Wu [9] has proposed an algorithm to examine neighbor pixels based on a decision tree. This algorithm also maintains an equivalence array similar to Suzuki's algorithm, to store label equivalence information.

As the second strategy of improving the speed, Kesheng Wu [9] has implemented Union-Find algorithms with an array rather than pointers. As it uses less memory he claims that, when all data structures fit into memory, the labeling algorithm with array based Union-Find is about five times faster than the similar algorithms with pointers.

By combining these two strategies, the algorithm fully captures the equivalence information only scanning the image once. After assigning provisional labels Fiorio's [3] algorithm has been used to assign final labels to all the connected components. This algorithm ensures that every root of connection tree has assigned minimal label and also assigns consecutive final labels to the components in one single pass through the equivalence array. According to the experimental results obtained by Kesheng Wu [9] for binary images, this algorithm has performed faster than all the other exiting algorithms for binary images.

In this section, only a brief description of each algorithm is given. Reader is encouraged to refer original papers for more details of these algorithms.

3. Changes for Above Algorithms to adopt them for Grayscale Images

Some slight changes were done for above algorithms to adopt them for grayscale images, directly without converting into binary images. First, a threshold value was defined to distinguish between the region of interest (ROI) and the background area of the image. All pixel values greater than the threshold value, are considered as belongs to the region of interest or else belongs to the background. Then if the value of the current pixel belongs to the background region, the mask (as shown in the figure 1) is moved to the next scanning position. If the current pixel belongs to ROI and all the other 4-nbr positions belong to background region or not equals to the pixel value of the current pixel then a new label to the current pixel is assigned. If two or more of neighbor pixels equal to the current pixel value assign the minimum label to the current pixel and mark other labels in 4-nbrs which the pixel value equals to the current pixel value, as equivalent. It has been used a separate matrix to store label equivalence information as used for binary images. Also it has been used a separate matrix to store assigned label information.

Furthermore, relevant changes were introduced for above original algorithms to compare gray values rather than comparing whether pixel values are one or zero.

4. Experiment

Goal of this experiment is to measure the performance of three algorithms (Park's algorithm, Suzuki's algorithm and Kesheng Wu's algorithm) on grayscale mammogram images without converting them into binary images. For this experiment 30 images were used from MIAS Mammographic database. Mammogram images in this database have been digitized at 200 micron pixel edge and clipped or padded; so that every image is of size 1024x1024 pixels. All the images in this database were stored as *portable gray map* (pgm) images. Above three connected component labeling algorithms were processed on 30 images, selected from MIAS database and measured processing time in seconds to complete the labeling for each image. Performance time was measured on grayscale image itself and corresponding binary image converted on the same threshold value used to distinguish ROI and background of the image. The algorithms were implemented on MATLAB. MATLAB uses a function called "im2bw" to convert grayscale images into binary images. In this function we need to specify the required intensity threshold value to convert from grayscale to binary image. This intensity value should be double precision value in the range of [0 1]. To keep the uniformity of the algorithm, intensity values of mammogram images were converted into double precision values that lies in the range [0 1]. Intensity value 0.2 is used as the threshold value to distinguish ROI (breast area) and background as on average this value covers the breast area on digital mammograms. Also these algorithms were tested on a Pentium 4 machine with 2.8 GHz speed and 512 MB memory.

5. Analysis of Experimental Performance

As explained above 30 images from MIAS database were tested for the performance of three connected component algorithms. As the first step of this analysis converted those mammogram images into binary images and compared the performance of these three algorithms. Even though comparisons about these algorithms had been done by previous authors; again we did the comparison with the aim of comparing the results with grayscale images based on same working conditions.

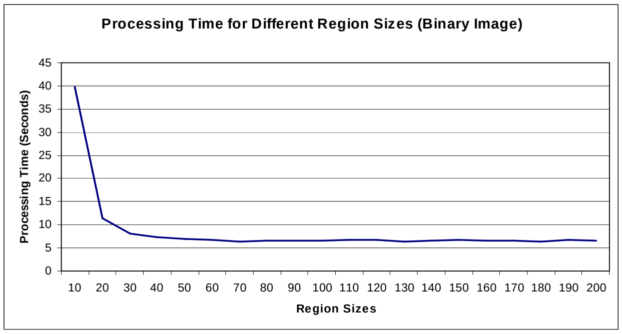

Fig. 3: Comparison of average processing time vs region size in Park's algorithm.

First, CPU processing time against the different region sizes of the Park's algorithm was compared. The results (average processing time vs. region size) are shown in the figure 3.

From this comparison we found that there is no significant difference on processing time for region sizes grater than 70x70 pixel regions on binary mammogram images. According to these results the average processing time for 1024x1024 binary mammogram images with 70x70 pixel region sizes will be around 6 seconds. Hence for further comparison with the performance of above three algorithms, region size 70x70 was used for the Park's algorithm. However, the average processing time proportionally depends on total number of pixels in connected components.

Figure 4 shows the performance comparison of the above three algorithms for binary images using MIAS mammogram image database.

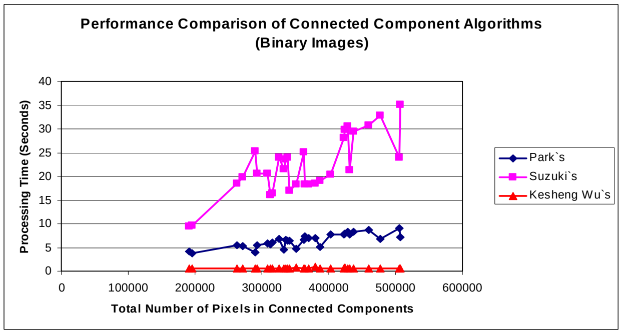

Fig 4: Comparison of CPU processing time of the three algorithms.

It clearly shows that there is a clear different between three algorithms. Kesheng Wu's algorithm clearly shows the lowest processing time; on average 0.58 seconds to recognize connected components in a 1024x1024 pixels mammogram image. Parks's algorithm performed connected component labeling with moderate time duration, while Suzuki's algorithm performed comparatively very high time duration.

Figure 5 and 6 show performance of modified version (for gray scale images) of Park's algorithm for different region sizes.

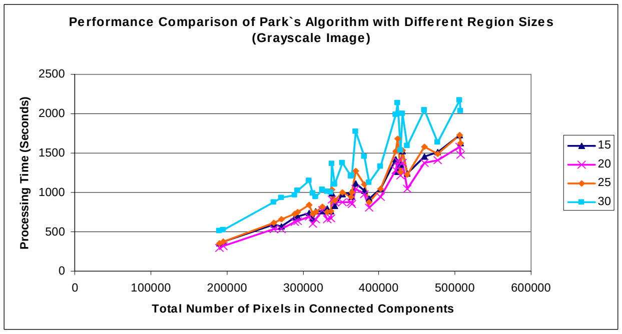

Fig 5: Performance comparison of Park's algorithm (modified for grayscale images) with different region sizes.

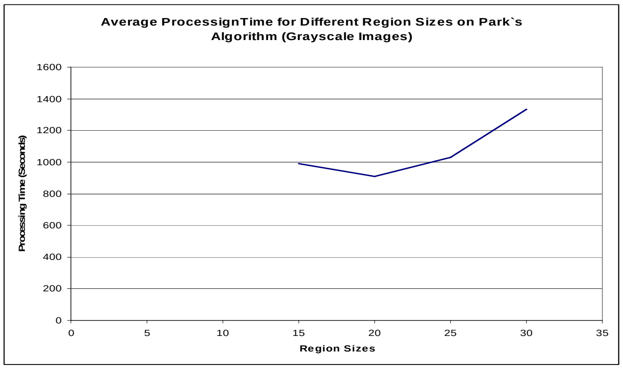

Fig 6: Average processing time for different region sizes on modified verion of Park's algorithm for grayscale images.

As figure 5 and 6 show, region size 20 gives optimum performance time for Park's algorithm on grayscale images. Hence the region size 20, is used to compare the performance of Park's algorithm with other algorithms for grayscale images.

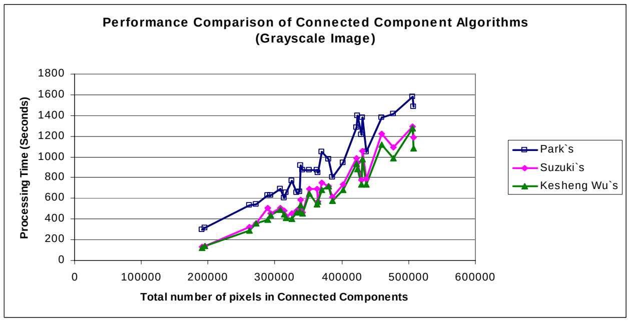

Fig 7: Performance comparison of connected component algorithms on gray scale images.

Figure 7 shows performance time comparison between the three algorithms for grayscale images. Even though Suzuki's and Kesheng Wu's algorithms showed a significant different on processing time for binary images, statistically there is no significant different on processing time between these two algorithms for grayscale images. CPU processing time statistics of these algorithms can be summarized and shown in table 1.

Table 1: Summary statistics of processing time for the three algorithms.

	Processing Time (Seconds)			
Algorithm	**Min**	**Max**	**Average**	**Std. Dev**
Park's	295.72	1581.82	910.77	348.19
Suzuki's	133.94	1297.17	665.50	303.17
Kesheng Wu's	126.49	1274.50	619.21	283.53

This comparison shows very high difference on processing time between binary images and grayscale images. Table 2 shows the comparison of total processing time for all 30 mammogram images used. On average Keheng Wu's and Suzuki's algorithms take 600 seconds per mammogram with moderate number of (350,000) pixels in connected component on a 1024x1024 pixel image. But the Park's algorithm takes comparatively higher duration for processing on grayscale images.

Table 2: Total processing time for all 30 mammogram images and ratio of processing time between binary images and grayscale images.

	Total Processing Time for all 30 images		
Algorithm	**Grayscale**	**Binary**	**Ratio**
Park	27323.10	193.55	141.17
Suzuki	19965.14	666.57	29.95
Keheng Wu	18576.34	17.62	1054.28

However these processing times showed in grayscale images are prohibitively high and would not be used for applications, especially in the area of medical image processing and diagnosis systems; because those applications are expected to give highly accurate results in lesser time duration. But the results given by this comparison would not be permitting us to recommend any of these algorithms to be used directly on grayscale images.

Further analysis, reveled several reasons for long CPU processing time. According to detail analysis, Park's algorithm has used about 35% of its processing time on equivalence relations resolutions. As the image size increases the size of the equivalence array also will be increased. This can be considered as the main reason for the high processing time. Lumia R.,[11] also has discussed this matter as a cause for increasing the processing time unexpectedly, referring to the [Rosenfeld and Pfalts] original algorithm. But in grayscale images the geometrical complexity of the image is very much higher than for binary images. Also in digitized mammogram images (grayscale) large fraction of connected regions are single pixel regions. This will increase the number of distinct regions on the image and hence further increase the size of the equivalence array. Park's algorithm uses searching algorithms to resolve equivalences and its time complexity is $O(K^3)$ for a *KxK* equivalence matrix. These bottlenecks increase the processing time of the Park's algorithm.

Initial results drawn on this algorithm was very much higher than the results shown above. But using an optimization technique in Matlab called vectorization we were able to decrease the processing time on average 40 ~ 50%. Operations that need "for" loops or "while" loops can be given as vector or matrix operations in MATLAB and they run faster than "for" loops or "while" loops.

Suzuki's algorithm and Kesheng Wu's algorithm used one dimensional table (*label connection table*) to store label equivalence information. Initially, the number of distinct labels that will be assigned to a given digital mammogram image is unknown. Hence, in the MATLAB implementation, we gradually increased the size of the label connection table as the algorithm recognized a new label for the image being processed. We found that these operations on *label connection table* consume a large fraction of total processing time. Later we found that every time as we increase the array size in MATLAB to accommodate new label names, MATLAB allocates a new memory block that fit in to required size and copies all previous values into this newly allocated memory block. This takes longer time and also decreases the performance considerably. The only solution is to overcome this bottleneck is to pre-allocate memory required *label connection table*. Pre-allocation also helps to reduce memory fragmentation when working with large matrices. In the course of a MATLAB session, memory can become fragmented due to dynamic memory allocation and de-allocation. This can result in plenty of free memory, but not enough contiguous space to hold a large variable. Pre-allocation helps prevent this by allowing MATLAB to "grab" sufficient space for large data constructs at the beginning of a computation [12]. But the main problem we faced here was to determine the size of the *label connection table* as we don't have information about the number of distinct regions in the image.

Results of the analysis revealed that there is a linear relationship between total number of pixels in ROI (or pixels in connected components) and total number of regions for given images (figure 8). This shows that total numbers of distinct regions for these given mammograms are approximately half the number of total pixels in ROI. Currently we are not in a position to generalize these results for all the cases; Even though there is no logical reason for this relationship, this would be a good topic for further analysis on all the available digital mammograms; as the given mammograms holds this relationship with very high precision.

However we adopted this relationship to these algorithms to test the improvement of the CPU processing time on Suzuki's algorithm and Kesheng Wu's algorithm. Initially we performed a single pass through a mammogram image and calculated the total number of pixels in ROI based on user defined threshold value to distinguish background and ROI regions.

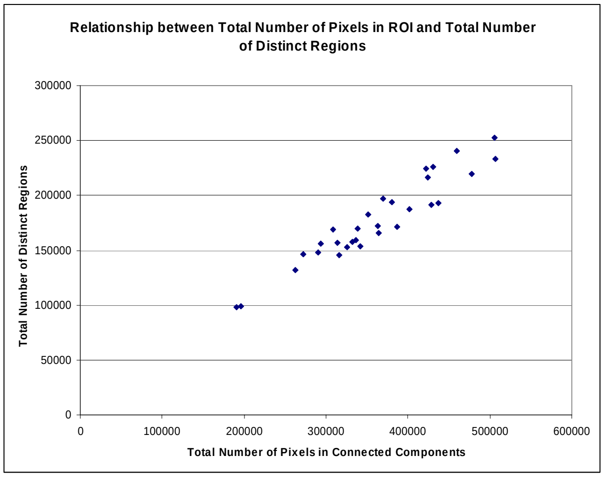

Fig 8: Relationship between total number of pixels in ROI (pixels in connected components) and total number of distinct regions for 30 mammograms used for the analysis.

Then pre-allocated memory for the label connection table with the size equals to half of the total number of pixels in the ROI. Then as assigning labels for connected components, if the next label value to be assigned exceeds the array size then the array size is increased by a constant value. This process reduces the number allocations and de-allocations of memory blocks by MATLAB and improved the CPU processing time dramatically (figure 9).

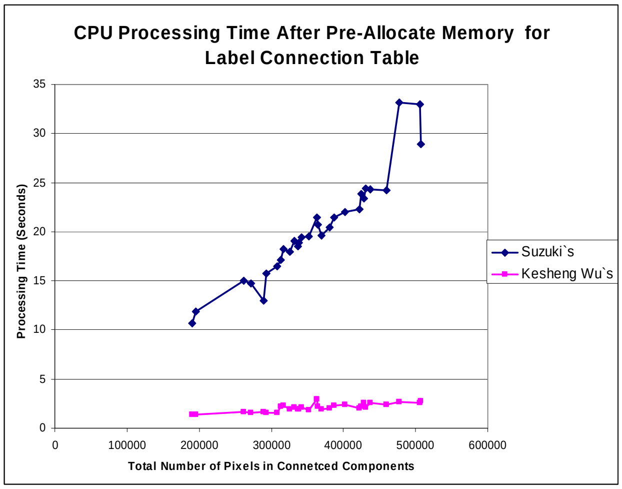

Fig 9: relationship between CPU processing time after pre-allocate memory for label connection table and total number of pixels in connected components.

This shows highly significant improvement compare to previous results. Also this shows significant difference between Suzuki's algorithm and Kesheng Wu's algorithm on processing time. Kesheng Wu's algorithm which shows the highest performance among these tree algorithms, varies its processing time from 1.34s to 2.94s while Suzuki's algorithm varies from 10.72s to 33.15s. Hence Kesheng Wu's algorithm performs approximately 5 to 15 times faster than Suzuki's algorithm. Kesheng Wu's algorithm completes the connected component labeling with one scan on image and one scan on label connection table and finally, one scan on label matrix to assign final labels in label connection table to label matrix. [Suzuki et al] has claimed that his algorithm completes connected component labeling by no more than four scans based on results evaluated on binary images. According to his results only 5 images among 2050 binary images needed four scans and all the others had complete labeling with only three scans. But according to the results obtained from our experiment it shows that, majority of images needed four scans to complete the connected component labeling (only one image completed labeling with three scans and two images required five scans).

Figure 10 and table 3 show summary statistics of performance time of Suzuki's and Kesheng Wu's algorithms on binary and grayscale images. Even though the graph (fig. 10) shows a slight increment in processing time for grayscale images in suzuki's algorithm; by analyzing these results statistically, it can be concluded that there is no significant different on processing time of Suzuki's algorithm for binary and grayscale images. But the Kesheng Wu's algorithm is approximately five times slower for grayscale images than binary images. But overall performance of modified version of Kesheng Wu's algorithm is higher than the Suzuki's algorithm.

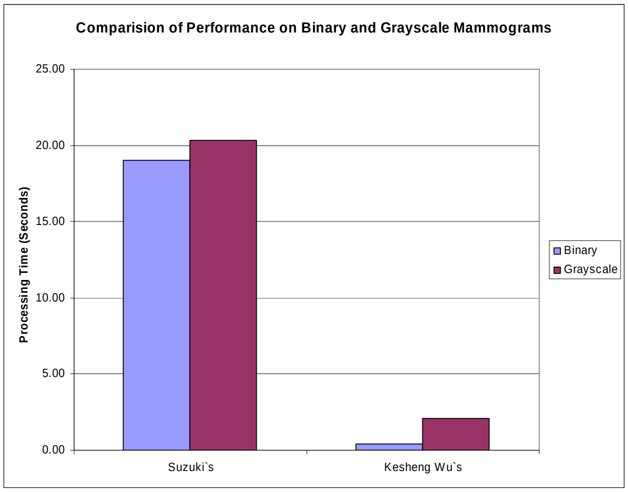

Fig 10: Comparison of performance time of Suzuki's and Kesheng Wu's algorithms on binary and grayscale images.

Table 3: Summary statistics of Suzuki's and Kesheng Wu's algorithms on binary and grayscale images.

		Processing Time (Seconds)			
	Algorithm	Min	Max	Avg	Std Dev
Binary Images	Suzuki's	9.52	35.03	22.22	6.20
	Kesheng Wu's	0.51	0.97	0.59	0.09
Grayscale Images	Suzuki's	8.51	30.54	19.00	5.28
	Kesheng Wu's	0.35	0.46	0.40	0.03

6. Conclusions

In this paper we compared three algorithms which was originally tested and evaluated on binary images and assessed the performance of them on grayscale images with necessary modifications. Especially on digitized mammograms. Using connected component algorithms directly on grayscale images would increase the performance and accuracy of computer vision applications such as medical image analysis and computer aided diagnosis. If we use connected component algorithms directly on grayscale images, it preserves information such as unknown patterns hidden in medical images. These information can be extracted and stored in data structures for further processes, while connected component labeling is being processed. This would be very much useful in multi-scale images processing and constructing scale-space primal sketch to store required information for further processing [10]. But this is impossible after converting grayscale into binary images. It causes greater information loss. Also this simplifies pre-processing by eliminating grayscale to binary conversion steps at different threshold values and eliminating adaptive threshold selection algorithms from the process.

Kesheng Wu's algorithm with above mentioned modifications shows fairly significant performance on mammography images. But still we can't generalize these results as we implemented and tested on MATLAB. It needs further analysis by implementing these algorithms on other standard languages and evaluation of results for generalization. We will work further on these algorithms for further improvements on processing time by doing necessary modifications.

References

[1] Christoyianni I., Dermatas E., Kokkinakis G., Automatic Detection of Abnormal Tissue in Mammography, In Proc. International Conference on Image Processing, Volume 2, p 877-880, 2001.

[2] David R., Arnau O., Joan M., Marta P., and Joan E., Brest Segmentation with Pectoral Muscle Suppresion on Digital Mammograms, In Proc. Iberian Conference on Pattern Recognition and Image Analysis(2), p 471-478, 2005.

[3] Fiorio C., and Gustedt J., Two Linear Time Union-Find Strategies for Image Processing, Theoretical Computer Science, Vol 154(2), p 165-181, 1996.

[4] Gonzalez R.C., and Woods R.E., Digital Image Processing (Second Ed), Prentice Hall, 2002.

[5] Haralick R.M., Some Neighborhood Operations, Real Time/Parallel Computing Image Analysis, Plenum Press, New York, p 11-35, 1981.

[6] Hashizume A., Suzuki R., Yolouchi H., An Algorithm of Automated RBC Classification and its Evaluation, Bio Medical Engineering, vol. 28(1), p. 25-32, 1990.

[7] Jasjit S., Suri, Rangaraj M., Rangayyan, Recent Advances in Breast Imaging, Mammography, and Computer-Aided Diagnosis of Breast Cancer, SPIE Publication, 2006 (In Print)

[8] Jung-Me P., Carl G. Looney, Hui-Chuan C., Fast Connected Component Labeling Algorithm Using a Divide and Conqure Technique, CATA 2000 Conference on Computers and Their Applications, pp 373-376, Dec. 2000.

[9] Kesheng W., Ekow O., and Arie S., Optimizing

Connected Component Labeling Algorithms, In Proceedings of SPIE Medical Imaging Conference, p. 1965-1976, Apr 2005.

[10] Lindeberg T., Scale-Space Theory in Computer Vision, Kluwer Academic Publishers, 1994.

[11] Lumia R., Shapiro L., Zuniga O., A New Connected-Components Algorithm for Virtual Memory Computers, Computer Vision Graphics and Image Processing, Vol 23, p 207-217, 1983.

[12] MATLAB User Manual

[13] Michael A., Wirth, A Non-rigid Approach to Medical Image Registration: Matching Images of the Breast", Ph.D. Thesis, 2000, RMIT University, Melbourne, Australia

[14]] RadiologyInfo, http://www.radiologyinfo.org/content/mammogram.htm

[15] Rosenfeld A., Pfaltz J.L., Sequential Operations in Digital Processing, Journal of ACM, Vol. 13, 471-494, 1966

[16] Suzuki K., Isao H., and Noboru S., Linear-Time Connected-Component Labeling Based on Sequencial Local Operations, Computer Vision and Image Understanding 89(1), p 1-23, Acedemic Press, 2003.