International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:9, No:11, 2015

Detecting Circles in Image Using Statistical Image Analysis

Fathi M. O. Hamed, Salma F. Elkofhaifee

Abstract**—**The aim of this work is to detect geometrical shape objects in an image. In this paper, the object is considered to be as a circle shape. The identification requires find three characteristics, which are number, size, and location of the object. To achieve the goal of this work, this paper presents an algorithm that combines from some of statistical approaches and image analysis techniques. This algorithm has been implemented to arrive at the major objectives in this paper. The algorithm has been evaluated by using simulated data, and yields good results, and then it has been applied to real data.

Keywords**—**Image processing, median filter, projection, scale-space, segmentation, threshold.

I. Introduction

IMAGE analysis combines techniques to meaningful information from image, in particular from digital image. Early work in statistical imaging was carried out [1], [2].

An object in an image could be described by a set of geometric structures such as lines, circles etc. These objects must be determined to give an aid in many practical sciences, such as, medical, geology, physics, engineering, etc. This paper proposes an algorithm that will be used to detect circles in image. This algorithm is a combination of methods. The first suggested method is thresholding method, which enables us to select ranges of pixel values in Gray-scale and colour image that separate the objects from the background. In this study, the threshold method is used to extract the foreground from the background in the image. Secondly, scale-space method is applied to smooth the density curve of thresholded image. This method is needed to find optimum bandwidth (smoothing parameter) which is used to find optimum smoothing density curve. Then the optimum thresholding is calculated from the optimum density curve. The optimum thresholded image may include some noise such as isolated pixels, to remove these unwanted pixels median filter approach has been proposed. Then projection method will be used to detect the main characteristics, which are locations, sizes, and number of objects in the image. In case of multi circles in the image, we suggest to split the main image into sub-images. Then the algorithm will be applied into each sub-image to detect the circles. After that the reconstructed image can be obtained by combine the detected sub-images.

F. M. O. Hamed and S. F. Elkofhaifee are with the Department of Statistics, Faculty of Science, Benghazi University, Benghazi- Libya; (e-mail: felramly@gmail.com).

II. Methods in the Algorithm

A. Thresholding Method

Thresholding is the simplest method of image segmentation. Thresholding often provides an easy and convenient way to perform segmentation on the basis of the different intensities or colours of the foreground and background region of an image. Reference [3] had used threshold to segment region to regions. Reference [4] uses thresholding to segmentation the soil image into different regions, and thresholding used to classification

B. Scale-Space Method

Scale-Space method had introduced in 1983 by Witkin [5], who proposed the term "Scale-Space". The Scale-Space is simply a family of Gaussian curve smoothers indexed by bandwidth (smoothing-parameter), sometimes there is important information available at several levels of smoothing parameter, when it is decreased the curve appears "undersmoothing", and as smoothing parameter is increased, curve appears "over smoothing" (see [6] for more details). Reference [7] used scale-space method to find optimum smoothing parameter. Reference [8] used the scale space texture classification to ultrasound liver tissue characterization for discrimination of normal liver from cirrhosis yields promising results.

The scale-space in this paper has been used to determine the optimum bandwidth that will be used to smooth the curve. Fig. 1, presented by [7], shows the relation between the location of thresholds, on the y-axis and various bandwidths. The number and values of the thresholds are changed according to the bandwidth. As the bandwidth increasing the number of threshold decreasing. The optimum number of the threshold based on number of the threshold that remains longer. Thus, the optimum bandwidth is selected to be in the middle of the threshold, that staying longest.

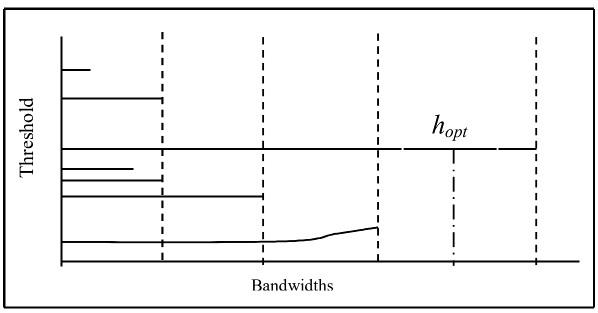


Fig. 1 Scale-space method Diagram

C. Median Filter Method

Median filtering in signal processing has been established by [9], and it has now become a standard technique in image processing. The median filter used extensively for image noise reduction and smoothing, it is especially good for removing impulsive noise.

In the binary image, some isolated pixels (unwanted pixels) may appear. For instance, white pixels in black region and black pixels in white region. This is called salt and pepper noise. The median filter method is proposed to deal with this kind of noise. Reference [9] used it to remove isolated pixels, [10] introduced a new solution for impulsive noise detection in colour images.

Reference [11] described the definition and testing of a new type of median filter for images. The new type is topological median filter and its statistical properties are estimated. [12] presented a basic tool for analyzing structures at different scales.

III. Proposed Algorithm

The proposal algorithm is combined of methods. It consists of thresholding approach, which is used to separate the foreground from background; and scale-space method is adopted to determine the optimum bandwidth that used to obtain optimum number and value of the thresholding. Then the median filter method might be needed to remove salt and pepper noise. These methods are applied sequentially in the proposed algorithm; some computation steps have been added to achieve our main goal, which are determined numbers, sizes and locations of the objects in image. This algorithm will be evaluated by apply to a simulated data; if the good performance is achieved then it will apply to real data.

A. Algorithm Steps

Suppose that a noisy image**, I,** where contains **n** circles as objects.

1. Suppose that ***I*** is an image with Gray levels *L*= [0, 255], the thresholds could be obtained using these steps,
   - Find the histogram for the image intensities, and let $f(x)$ denote to the frequency, where x a sequence of Gray levels from 0 to 1.
   - Compute $g(x)$, where $g(x) = f(x+1) - f(x)$.
   - Observe the signs of each value in the vector $g(x)$, if $g(x)$ <0 and $g(x+1)$ >0 then the value of $x+1$ will be stored in vector *U*.
2. Use Scale-Space Algorithm to fined optimum smoothing parameter for Histogram curve as shown in these steps:
   - Create a sequence of values with initial bandwidth in middle. The difference between the values should as small as possible.
   - Each value in the sequence is used, as a bandwidth, to plot histogram, of the data. Then for every plotted histogram count number of minima (thresholds), as illustrated in first step and store the results in vector Y_T. Some of the minima are staying longer and this lead to repetition values in vector Y_T.
   - The median of these values, which are staying longest in the vector Y_T, is considered as the optimum bandwidth h_{opt} (say).
3. Apply the Median Filter Method, to remove isolated pixels which may appear in the thresholded image, suppose that the matrix of obtained image, $f(i,j)$.

The median filter considers each pixel in the image in turn and looks at its nearby neighbors to decide whether or not it is representative of its surroundings. It replaces the value of pixel by the median of the gray level in the neighborhood of that pixel: $f(i,j)$=median $\{I(i,j)\}$, where $f(i,j)$ is filtered image, $I(i,j)$ is the image. Now the mean square differences (*MSD*), are computed according to the following steps:

4. Set new matrix, ***y***, with zeros elements, $y(i,j) = 0$, $i = 1,2,..., N$. $j = 1,2,..., M$,
5. Compute MSD,

$$MSD = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\left(f(i,j) - y(i,j)\right)^2}{NM}$$

6. Set $i = 1$, $j = 1$ in $y(i,j)$
7. Set $y(i,j)$=1.
8. Recalculate MSD with new Value as:

$$MSD^* = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\left(f(i,j) - y(i,j)\right)^2}{NM}$$

9. Define another zeros matrix, ***x***, of size $N \times M$ and set $x(i,j)$=*MSD**
10. If *MSD*<MSD*, then set $y(i,j) = 1$ and *MSD=MSD** otherwise $y(i,j) = 0$
11. If $j < M$ then put $j = j+1$ and go to step (7)
12. If $i < N$ then put $i = i+1$ go back to step (7).
13. Calculate mean of each row in matrix x and store the results in vector R.
14. Set $i = 1$
15. If $R(i+1) < R(i)$, store i in vector A
16. If $(i+1) \le N$ set $i = i+1$ and go back to step 15
17. Set new vectors a, a^*, let $a_1 = A_1$, $a_l^* = A_k$ and, If $A_{k+1} - A_k > 1$, $k = 2,3...$, Length of (A), $l = 1,2,...$, length of (a) Store A_k in a, A_{k+1} in a^*

18. Calculate mean of each column in matrix $x(i,j)$ and store the results in vector C.
19. Set $j=1$
20. If $C(j+1)<C(j)$, store j in vector B
21. If $(j+1)\le M$ set $j=j+1$ and go back to step 20
22. Set new vectors b, b^*, Let $b_1=B_1$, $b_l^*=B_{k'}$ and, if $B_{k'+1}-B_{k'}>1$, $k'=2,3,...,$ length of (B) and $l=1,2,...,$ length of (b), Store $B_{k'}$ in b and $B_{k'+1}$ in b^*
23. Number of elements in vector a represents number of objects "circles" in image. Note that length of a is equal to length of a^* and length of b is equal to length of b^*.
24. Compute radii of circles , r_l, as following

$$r_l=\frac{a^*[l]-a[l]}{2N},\ r_l=\frac{b^*[l]-b[l]}{2M},$$

$$l=1,2,...,\text{ length } (a)$$

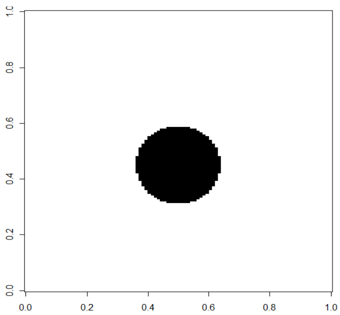
25. Calculate coordinates of the locations "centers" of circle c_l, c_l^*, where c_l is first coordinate, c_l^* is second coordinate $c_l=\frac{a_l+a_l^*}{2N}$, $c_l^*=\frac{b_l+b_l^*}{2M}$
26. Using r_l, c_l, c_l^* results to reconstruct circles.

IV. Applications

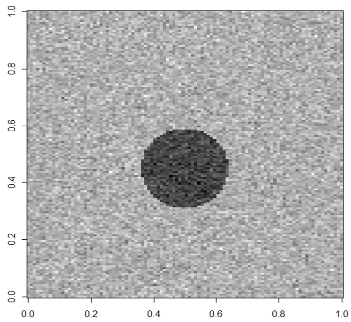
A. Application to a Synthetic Data

Suppose that one circle is simulated in 2-D, ***n*=1.** The center of this circle (x, y) is generated from the uniform distribution on the region (0,1), while the radius (r) is simulated from lognormal distribution with parameters, μ=-3, σ^2=0.1^2. That is $x\sim U(0,1)$, $y\sim U(0,1)$ and $r\sim \log N\ (-3, 0.1^2)$.

The simulated circle is converted to a binary image as shown in Fig. 2 (a). The resolution of the image is 150×100 pixels. The image is contaminated by Gaussian noise with mean equal to zero and standard deviation equal to 0.1, the noisy image is plotted in Fig. 2 (b).



(a)



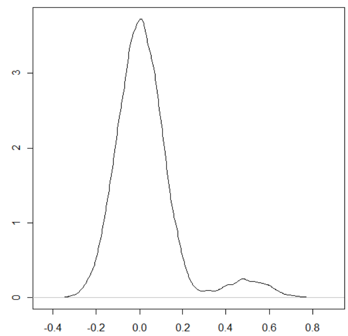
(b)

Fig. 2 (a) Image of one circle; (b) Noisy image

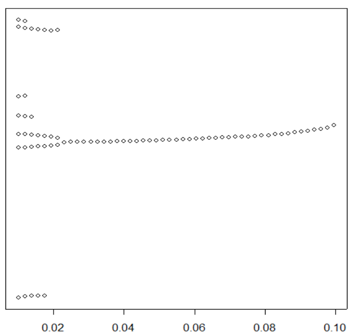
The density curve of the noisy image is obtained and plotted in Fig. 3 (a). Although some negligible minima, which are resulted from the noise, the density curve can be considered a bimodal curve, then one threshold is enough to segment the image.

The optimum smoothing parameter is needed to smooth the unimportant minima in the density curve, then the scale-space method will be used. Fig. 3 (b) shows the resulted graph of scale-space method. It yields optimum bandwidth equal to 0.061.

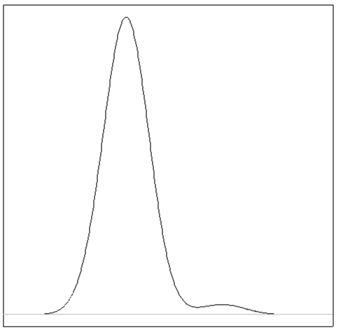
The optimum bandwidth has been used to find the optimum smoothing density as plotted in Fig. 3 (c), which shows the density curve with only one threshold at $T=0.36$.



(a)



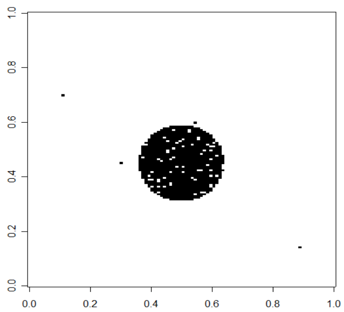
(b)



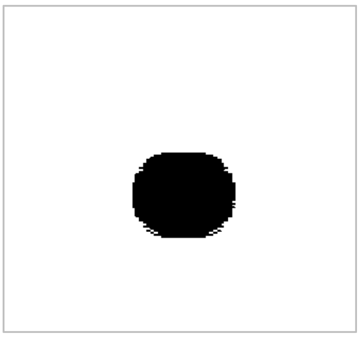
(c)

Fig. 3 (a) Density Curve; (b) Scale space diagram; (c) Optimum smoothing Density

The obtained threshold has been applied to the noisy image and the resulted thresholded image is plotted in Fig. 4 (a), it still contains some isolated pixels (salt and pepper noise). The isolated pixels can be removed via median filter method (step 3 in the algorithm). Fig. 4 (b) displays the filtered image after apply the median filter method.



(a)



(b)

Fig. 4 (a) Thresholded image; (b) Filtered image

The matrix $x(i, j)$ has been obtained (as in Steps 4-12). Mean of each row in the matrix $x(i, j)$ will be computed and stored in vector R . Fig. 5 (a) shows the plot of vector R .

The values in the vector R are constants at the rows, which have no objects and they will be decreased at the rows have pixels of foreground. The ranks of the rows of the decreasing values are stored in vector A as, A= (36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56,57, 58,59, 60, 61,62, 63, 64).

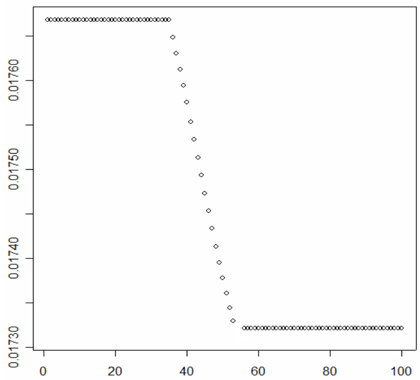
The values in vector A are successive values, that is, there is no gap between the values. This means the image has one circle. Likewise, the same procedure has been conducted on the columns in the matrix $x(i, j)$, the results of mean of each column are stored in vector C , Fig. 5 (b) shows these results.

The ranks of the decreasing values, in the vector C , are stored in vector B ,

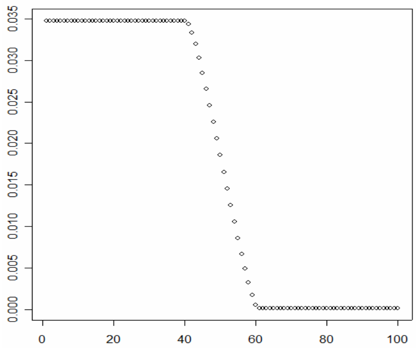
B =(47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69,70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88).

Once more, the vector B confirmed that the image has one circle. Radius of the circle, will be calculated as explained in step 24 in the general algorithm, the radius, r of the circle can be obtained by using the first and the last values in the vector A or B , $r_A = \frac{64-36}{2\times 100} = 0.14$ or $r_B = \frac{88-47}{2\times 150} = 0.14$,

Since the object in this example is a circle then r_A and r_B must be equal.



(a)



(b)

Fig. 5 (a) Mean of rows; (b) Mean of column

By applying step 21 in the algorithm, the center of the circle is equal to $(0.5, 0.45)$. The obtained results are: number of objects is one, the radius of the object is 0.14 and the location is at $(0.5, 0.45)$. According to these results, the object has been reconstructed as in Fig. 6.

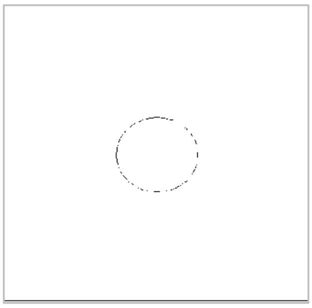


Fig. 6 Reconstruction dataset

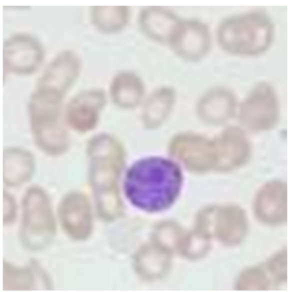
The result indicates that the algorithm yielded good achievement to identify one circle. This can be seen in this example, where the algorithm detects number, size, and location of objects in the image. This result encourages us to apply the algorithm on real data.

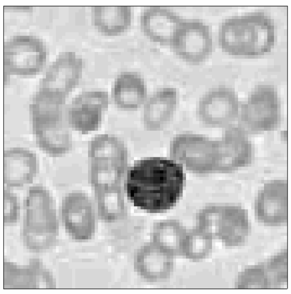
B. Application to Real Data

In this example, a slide of blood cell is used as a real data, Fig. 7 (a).

Since the interesting information can be obtained entirely from a grayscale representation, the data is converted to gray scale image. The grayscale image is shown in Fig. 7 (b) with 87×86 pixels, where $N = 87$ and $M = 86$. To segment the grayscale image, the histogram curve is plotted in Fig. 8 (a).

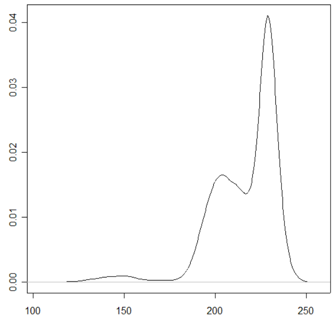
The optimum smoothing parameter is obtained by using scale-space method as in Fig. 8 (b), the arrow points to the optimum smoothing value, which is equal to 9.6. Fig. 8 (c) shows smoothing histogram, the value of thresholding can be seen in this histogram as a single threshold and it is determined at (T=163.72).



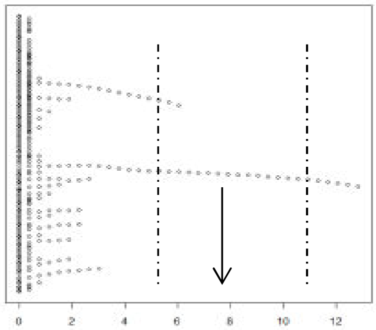


(a) (b)

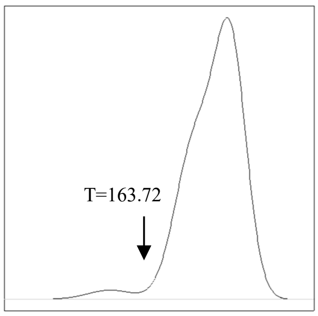
Fig. 7 (a) Slide of blood cell2; (b) Grayscale image of blood cell



(a)

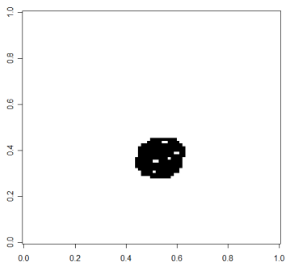


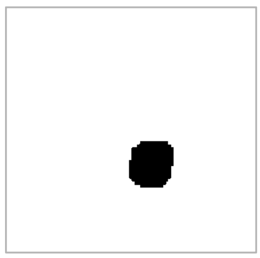
(b)



(c)

Fig. 8 (a) Curve for grayscale image, (b) Scale-space diagram, (c) Optimum Smoothing Histogram





(a) (b)

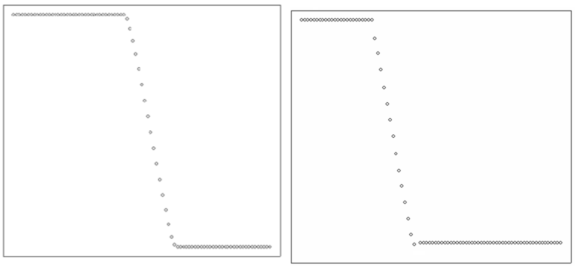
Fig. 9 (a) Thresholded image; (b) Filtered image

The grayscale image is segmented using the obtained threshold as in Fig. 9 (a). The thresholded image shows some isolated pixels, which can be removed via median filter method. The filtered image is displayed in Fig. 9 (b).

Fig. 10 (a) shows means of the rows of the filtered image $x(i,j)$. Fig. 10 (b) presents means of the columns in the filtered image $x(i,j)$. Both figures detect a single object in the image.

The different points in Figs. 10 (a) and (b) are stored in the vectors A and B respectively. The values in A represent the rows which contain object, while the values in B represent the columns have object,

$$A = (38,39,40,41,42,43,45,46,47,48,49,50,51,52,53,54,55),$$
$$B = (24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39).$$



(a) (b)

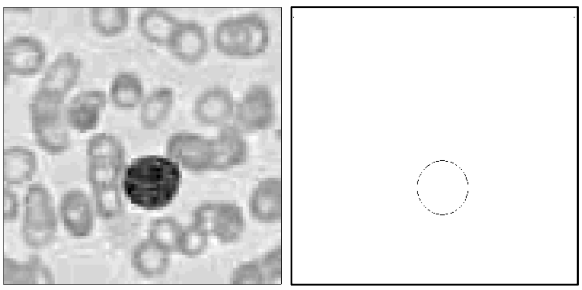
Fig. 10 (a) Mean of rows; (b) Mean of columns

The values in vector A and vector B are in sequence, without any gap, this means number of objects in the image is one.

The radius of the circle has been obtained as:

$$r_A = \frac{55-38}{2\times 87} = 0.09 \text{ or } r_B = \frac{39-24}{2\times 86} = 0.09$$

The center coordinate of the circle is found equal to $(0.53,0.37)$. By employing these results $n=1$, $r=0.09$ and center $=(0.53,0.37)$, the object is reconstructed in Fig. 11 (b).



(a) (b)

Fig. 11 (a) Grayscale image of blood cell; (b) Reconstructed image

The well identification of the object, in the real image, has been achieved by using the algorithm. Therefore, the algorithm works well in case of one circle in the image.

So far, the paper presented an algorithm and examples to explain how to detect the circles in image. This detection would be misleading, when a circle is shared with others in the rows or columns. In the other words if same row or same column bass through more than one circle the previous algorithm will identify these circles as one object. To resolve this matter the next section suggests an extra step to the algorithm.

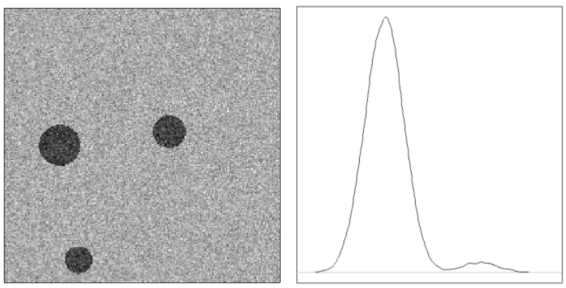
V. Multi-Circles Detection

This section proposes an approach to identify circles multi-circles in an image. The proposal based on divide image into sub-image. The sub-image contains non-overlapping objects images. The approach starts by run the algorithm till get the matrix MSD^* (steps from 1 to 12). Then the matrix will be used to divide the image. The partitions will be considered at the mid-distance between the overlapping objects. The procedure is explained in the following example.

A. Application to a Synthetic Data

The data in this example has been simulated with three circles. The centers of circles are generated from uniform distribution on the region $(0,1)$. The radii of circles $\{r_i, i=1,2,3\}$ are created from a lognormal distribution with parameters, μ=-3, $\sigma^2 = 0.1^2$. That is $x \sim U$ (0,1), $y \sim U$ (0,1) and $r_i \sim \log N$ (-3, 0.1). The circles in the simulated data are generated with overlapping in rows and columns. This data is converted to an image of size 200×150 pixels.

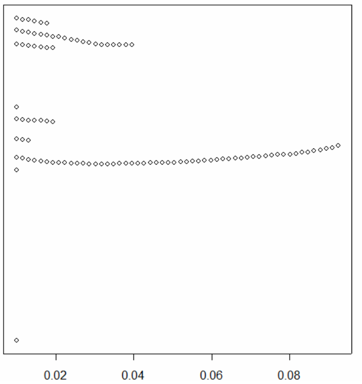
Fig. 12 (a) shows noisy image with $\varepsilon \sim N$ (0, 0.1). The algorithm has been implemented. The density of noisy image is plotted in Fig. 12 (b).



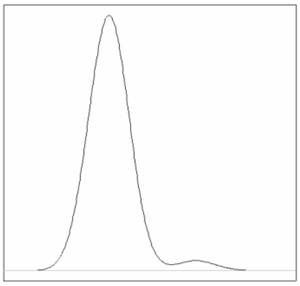
(a) (b)

Fig. 12 (a) Noisy image; (b) Density of noisy image

The scale-space method has been applied and shown in Fig. 13 (a). The optimum bandwidth found at 0.065. This value is used to plot the density in Fig. 13 (b). Then the threshold is determined at $T = 0.35$.



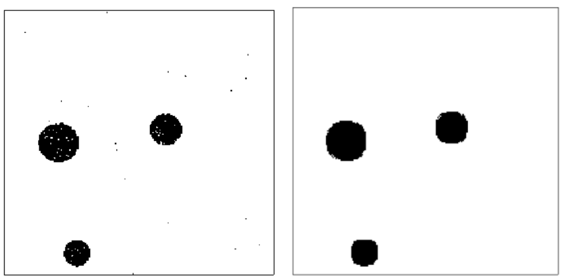
(a)



(b)

Fig. 13 (a) Scale-space method; (b) Optimum smoothing curve

The thresholded image is presented in Fig. 14 (a). The salt and pepper noise appear in the thresholded image, the median filter has been applied to remove this noise, the filtered image is shown in Fig. 14 (b).



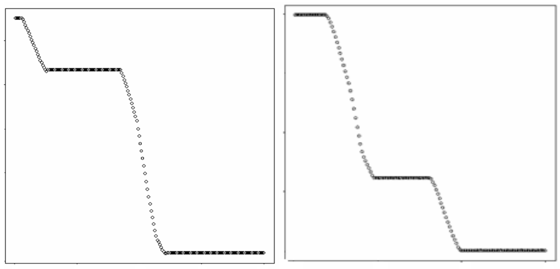
(a) (b)

Fig. 14 (a) Thresholded image; (b) Filtered image

The two graphs in Figs. 15 (a) and (b) show that number of objects is two. But the image has been simulated with three circles. This error in identify number of objects is occurred due to the overlapping in the image.

To solve this problem, new matrix, *d*, is defined. The elements in this matrix are resulted from subtract each value in MSD^* matrix from the previous value in each row,

$$d(i, j) = MSD^*(i+1, j) - MSD^*(i, j) .$$



(a) (b)

Fig. 15 (a) Mean of rows; (b) Mean of columns

The values in the matrix *d* will not be equal to zero when these values are corresponded to the pixels in the foreground while the rest values are equal to zero.

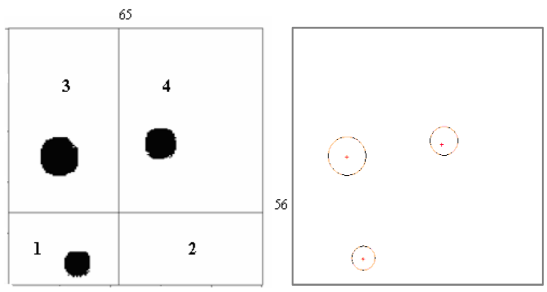
The orders of the rows for the non-zero values, in the matrix d, are stored in vector V,

V =(7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, **99,99, 100, 100,101,101, 102, 102, 103, 103, 104, 104, 105, 105, 106, 106, 107, 107, 108, 108, 109, 109, 110, 110, 111, 111, 112, 112, 113, 113, 114, 114, 115, 115,** 116,117, 118, 119, 120, 121, 122)

The repeated values in the vector V mean that there is an overlapping corresponding to these rows. The discontinuity in the values in the vector, $V = (26,86)$, determine the distance between the objects. The partition, between the overlapping objects, is considered at the row that in the mid-distance, that is, $\frac{26+86}{2} = 56$, the image will split into two image at row 56. The same procedure will be repeated on the columns and the resulted vector is,

V' =(20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, **34, 34, 35, 35, 36, 36, 37, 37, 38,38, 39, 39, 40, 40, 41, 41**, 42, 43, 44, 45, 46,47, 48, 82,83, 84, 85, 86, 87, 88, 89, 90,91, 92, 93, 94, 95, 96,97, 98, 99)

The partition at the columns is calculated as, $\frac{48+82}{2} = 65$, then the image will divide at the column 65. Fig. 16 (a) shows the filtered image with the partitions.



(a) (b)

Fig. 16 (a) Divide the image into sub-image; (b) Reconstruction data

The image is divided into four sub-images by using the obtained partition. The algorithm has been applied to find number, locations and size circles in each sub-image.

The algorithm yields that, three of the four sub-images possess one circle. The radii of the three circles are (0.05, 0.08, 0.06), and their centers are (0.27, 0.08), (0.2, 0.5), (0.61, 0.56) respectively.

The reconstruction data is aggregated and presented in Fig. 16 (b). This result shows that the approach gave good achievement, and the concept of sub-image helps to solve the problem of overlapping.

B. Application to Real Data

The Real image has intensity of 113×150 pixels. It is presented in Fig. 17.

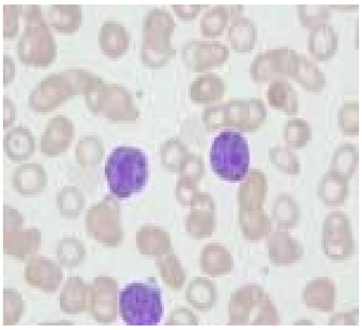
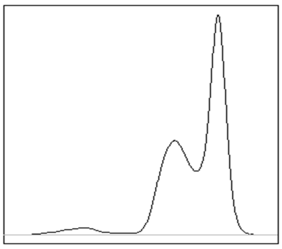
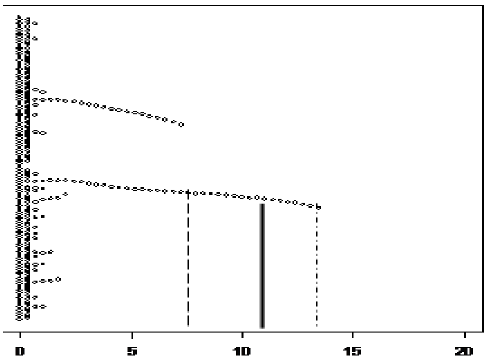


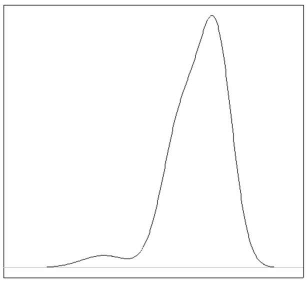
Fig. 17 Slide of the blood cells



(a)



(b)

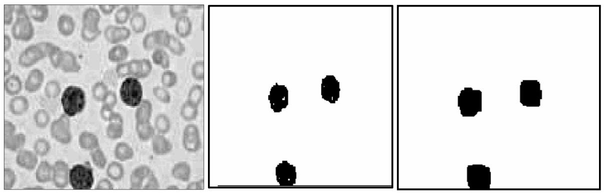


(c)

Fig. 18 (a) Histogram curve; (b) Scale-space method; (c) Smoothing Histogram curve

The histogram curve is plotted in Fig. 18 (a) and the result of the scale space method in Fig.18 (b). The obtained optimum bandwidth (10.5) is used to smooth the histogram as shown in Fig. 18 (c). Then the threshold is obtained at 169.47.

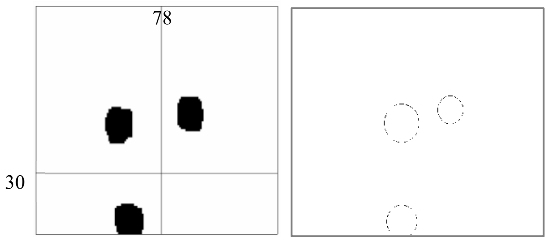
Fig. 19 summarizes the steps to get binary image. The true image is displayed in Fig. 19 (a), the thresholded image is plotted in Fig. 19 (b), and Fig. 19 (c) shows the filtered image.



(a) (b) (c)

Fig. 19 (a) Gray-scale image; (b) Thresholded image; (c) Median filtered image

Due to the overlapping between objects in the image. This will causes difficulties to identify the objects. The same approach, which explained in the previous application, will be applied to deal with overlapping issue. According to that approach, the filtered image will divide into four sub-images, as in Fig. 20 (a).



(a) (b)

Fig. 20 (a) Image to Sub-images; (b) Reconstruction of real data

The remaining steps of algorithm, to detect the objects, are applied to each sub-image. The algorithm detects one object in three sub-images, and non-object in one sub-image. The radii of the three circles found as (0.07, 0.08, 0.06), and their centers are (0.39, 0.07), (0.43, 0.48) and (0.65, 0.54) respectively.

The results used to reconstruct data as in Fig. 20 (b). This shows that the approach gives good performance, in case of real data.

VI. Conclusion

The aim of this paper was to detect geometrical objects in an image. The objective has been achieved via suggested algorithm.

The proposed algorithm, in this paper, has been evaluated by using a synthetic data and achieved good results. This achieved encouraged to apply the algorithm to real data. In case of real data, algorithm yields well result. The algorithm provided good results and achieved the aims of this research when the image contains one object, but such good results are not be easy when the objects in image are overlapped, in such case extra steps are needed to the algorithm.

References

[1] S. Geman and D. Geman, “Statistical analysis of dirty pictures,” in *Advances in Applied Statistics*, vol. 20, K.V. Mardia and G.K. Kanji, Ed. Leeds: UK, 1993, pp. 63–87.
[2] J. Besag, “On the statistical analysis of dirty pictures (with discussion),” *The Royal Statistical Society J.*, vol. 48, pp. 259–302, 1986.
[3] B. Jahne, *Digital image processing*. New York: Springer, 2003.
[4] C. A. Glasbey and G. W. Horgan, *Image analysis for the biological scienc*es. Chichester: Wiley and Sons, 1995.
[5] A. P. Witkin, “Scale-Space filtering,” *in proc. 8th international joint Conf. on Artificial intelligence*, Germany, 1983, pp. 1019-1022.
[6] J. S. Marron and P. Chaudhur, “Scale-Space view of curve estimation,” *The Annals of Statistics*, vol. 28, pp. 408-428, 2000.
[7] F. M. Hamed, “Geometrical modeling and identification of structure in image analysis*,” phD thesis*, University of Leeds, 2005.
[8] M. J. Gangeh, R. P. W. Duin, C. Eswaran and B. M. Rommeny, “Scale Space Texture Classification Using Combined Classifiers with Application to Ultrasound Tissue Characterization,” in *proc. Intern. Conf. on Biomedical Engineering*, 2006.
[9] J. K. Tukey, “Nonlinear (nonsuperposable) methods for smoothing data,” *in proc. Congr. Rec. EASCOM*, 1974, pp. 673-681.
[10] J. Lianghai and L. Dehua, “A switching vector median filter based on the CIELAB color space for color image restoration,” *Signal Processing*, vol. 87, pp. 1345-1354, 2007.
[11] H. G. Senel, R. A. Peters and B. Dawant, “Topological Median Filters,” *in IEEE Transactions on Image processing*, vol. 11, pp. 89-104, 2002.
[12] T. Lindeberg, “Scale-Space theory: A basic tool for analyzing structures at different scales,” *Statistics and Images*, vol. 21, pp. 225-270, 1994.