

Available online at www.qu.edu.iq/journalcm

JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS

ISSN:2521-3504(online) ISSN:2074-0204(print)



Memory Organization for Image Segmentation Based on Region Growing

Osama Majeed Hilal Almiahi ^a

^aUniversity of al-qadisiyah, Ad Diwaniyah, Iraq.Email: osama.m@qu.edu.iq

ARTICLE INFO

Article history:

Received: 02 /01/2022

Revised form: 05 /02/2022

Accepted : 22 /02/2022

Available online: 28 /02/2022

Keywords:

FIFO-stack

Image segmentation

Memory organization

Region growing

ABSTRACT

A solution is proposed for determining and organizing the size of the FIFO-stack in memory, required for load the coordinates of adjoining pixels in an image segmentation algorithm based on region growing. The FIFO-stack, organized on the idea of a ring multi-bit shift register, is considered, and the expression of memory calculations is performed through storing process in parallel registers. The conditions for maximum loading of the memory stack are formulated, for which an expression is obtained that allowing to accurately determining the required size of the FIFO-stack, which provides memory resources savings.

MSC.41A25; 41A35; 41A3

<https://doi.org/10.29304/jqcm.2022.14.1.880>

1. INTRODUCTION

Memory allocation is one of the main problems of organizing data processing in computer systems. This problem is particularly related for systems on a chip with a limited size of RAM [1-4]. Insufficient memory allocation can lead to data loss and system malfunction. Excessive memory allocation leads to its inefficient use. Despite the lack of a universal approach to memory allocation, the exact values of its necessary capacity for specific algorithms under extreme conditions can be obtained. For avoiding limitation of memory resources, researchers developing an methods of image compression [5], while it's possible to reduce required memory by image pre-processing techniques. Among image pre-processing algorithms, segmentation algorithms are the most demanding of RAM capacity [6-9]. Segmentation result leads to the separation of an image into areas with the same or similar properties. The simplest segmentation algorithms are based on region growing in the surrounding area of the previously selected initial growth pixels and use stacks to store the coordinates of adjacent pixels attached to the grown region [10 – 12]. Stacks can be organized according to the FIFO principle (first in - first out) or LIFO (last in - first out) [13, 17]. They differ in the order of data sampling and, accordingly, in the memory capacity required to store the coordinates of pixels at maximum load. These applications of image processing and analyzing widely used in medical applications [19-21], remote sensing [22] and aerial images, which requiring high memory resources [14, 16]. Many of presented methods of image segmentation refers to allocating of segments and their characteristics,

* Corresponding autho : Osama Majeed Hilal Almiahi

Email addresses: osama.m@qu.edu.iq

Communicated by: Dr. Rana Jumaa Surayh aljanabi

such as edges, size and borders of segments in homogenous regions, regardless an important calculation factor of memory resources [23-25].

The aim of this work is to deriving an expression, which allows calculating the size of the FIFO stack, required for storing the coordinates of adjoining pixels depending on the image size for the segmentation algorithm based on region growing. The formatter will need to create these components, incorporating the applicable criteria that follow.

The structure of this paper as follows: Section 2, describes formulation of the problem; section 3, shows the derivation an expression that determining the calculation of memory; section 4 present the evaluation of memory savings for the exact calculation of the size of FIFO-stack. Section 5 concludes the conclusion.

2. FORMULATION OF THE PROBLEM

As a result of image segmentation $I = \left\| i(y, x) \right\|_{(y=0, \overline{Y-1}, x=0, \overline{X-1})}$ with sizes $Y \times X$ pixels forming the matrix $S = \left\| s(y, x) \right\|_{(y=0, \overline{Y-1}, x=0, \overline{X-1})}$ of segmentation having the same size, the values of the elements which indicate the number $n_s \in [0, N_s]$ of segments, which they belong, where N_s -number of segments and number of last segment. On Fig. 1 showing the fragment of the halftone image, the matrix of values of pixels corresponding to it and the segmentation matrix containing six segments are given.

Algorithm of region growing [15 - 18] describes the following sequence of operations in image segmentation, assuming that the initial pixels of growth in the image are already allocated [8, 16]:

1) Coordinates of the next initial pixel of growth placed in FIFO stacks $Y_F = \left\| y_F(p_F) \right\|_{(p_F=0, \overline{P_F-1})}$ and $X_F = \left\| x_F(p_F) \right\|_{(p_F=0, \overline{P_F-1})}$ coordinates of adjacent pixels, where p_F - pointer of stacks; P_F - maximum possible number of coordinates in FIFO stacks Y_F and X_F ;

2) determining the current value n_s as number of segment;

3) Implementing the processing cycle of FIFO-stacks Y_F and X_F , in which coordinates are retrieved (y, x) of next processed pixel; corresponding element of the segmentation matrix $s(y, x)$ assigned value n_s number of segment; checks satisfying the condition of joining the current pixel (y, x) adjacent pixels with coordinates $(y + y', x + x')$ when $(y' = \overline{-1, 1}) \wedge (x' = \overline{-1, 1}) \wedge ((y' \neq 0) \vee (x' \neq 0))$; if the condition is satisfied for any adjacent pixel, then its coordinates $(y + y', x + x')$ recorded in the stacks Y_F and X_F ;

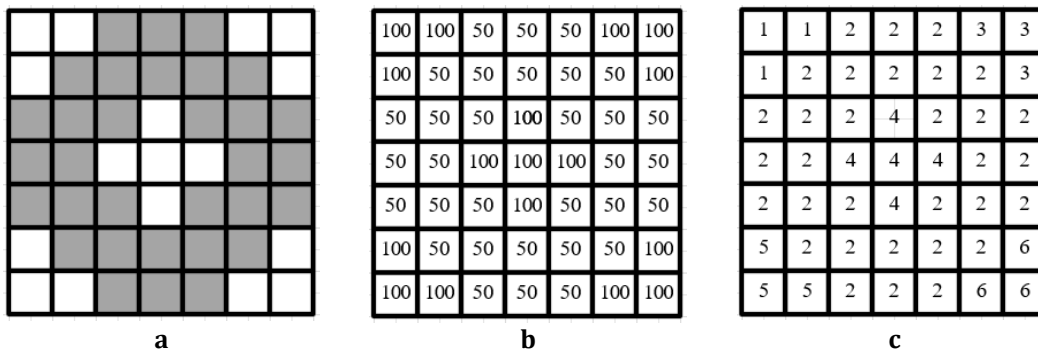


Fig. 1- Fragments: a - grayscale image; b - matrix of pixels; c - matrix of segmentation

4) Steps 1 – 3 repeated as long as all the initial pixels of growth have been processed.

FIFO- stacks Y_F and X_F having the same structure, shown at Fig. 2. It includes:

- circular multi-bit shift register, consisting of P_F identical cells, each of which contains a parallel register (RG) for storing the coordinate value and multiplexer (MX) for loading into the parallel register the values of the coordinates from the output of the previous one along the ring of the parallel register or from the external input;
- input binary counter (CT2) by module P_F , forming a pointer value p_{FE} at the end of the data in the FIFO stack, and descrambler (DC), selecting a shift register cell depending on the pointer value p_{FE} for loading the coordinate value to the FIFO-stack entry;
- output binary counter (CT2) by module P_F , forming a pointer value p_{FB} to the beginning of the data in the FIFO stack, and the multiplexer (MX), transmitting to the output of the FIFO- stack the value of the coordinate from the cell of the shift register, selected depending on the value of the pointer p_{FB} .

When loading into the FIFO-stack, a "Load" signal is generated, on the front of which the value of the pixel coordinates from the stack input is written to the current cell of the shift register, determined by the pointer value p_{FE} , formed by the input counter CT2. According slice signal "Load" value p_{FE} input counter CT2 increasing by one according to the expression $p_{FE} = \text{mod}_{P_F}(p_{FE} + 1)$. When unloading from the FIFO stack, the "Unload" signal is generated, on the front of which at the stack output, the value of pixel coordinates is set from the output of the current cell of the shift register, determined by the pointer value p_{FB} , generated by the output counter CT2. According slice signal "Unload" value p_{FB} output of counter CT2 increasing by one according to the expression $p_{FB} = \text{mod}_{P_F}(p_{FB} + 1)$. FIFO-stack is empty, when $p_{FE} = p_{FB}$. When initialized FIFO-stack input and output binary counters are reset by the «Reset» signal.

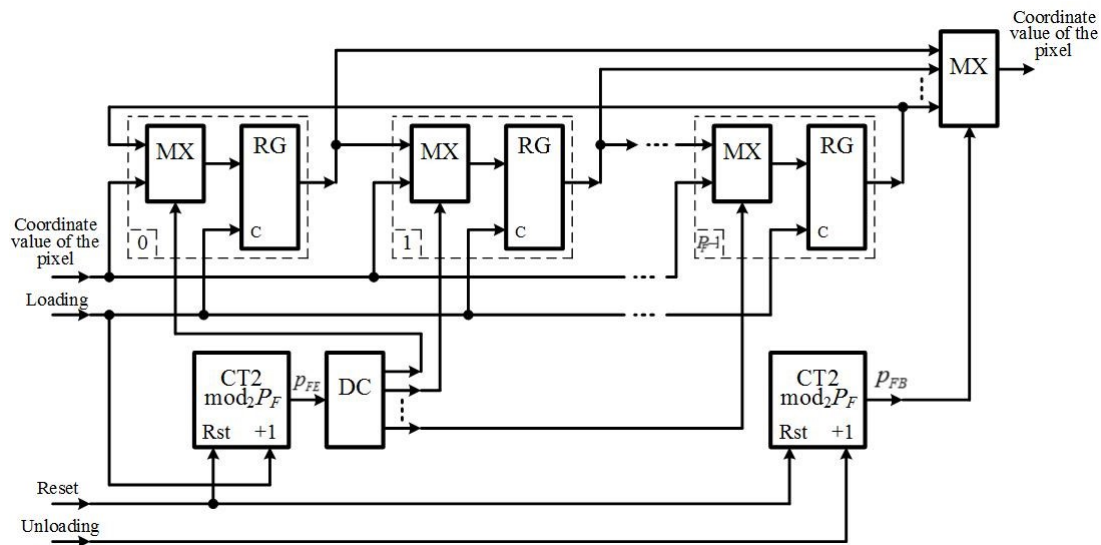


Fig. 2- Structure FIFO-stacks and with coordinates of adjacent pixels

Loading FIFO-Stacks Y_F and X_F maximum when the size of the segment coincides with the size of the image. In this case $N_s = 1$, $\forall y \forall x (s(y, x) = N_s)$, the size $R_s(n_s)$ of n_s - segment, defined by the expression $R_s(n_s) = \frac{1}{n_s} \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} s(y, x)$, makes up YX ($R_s(1) = YX$) with the exception of re- entering the coordinates in the stacks. In the absence of an expression to accurately determine the size of the stack (value P_F the number of cells), ensure the stable operation of the region-growing algorithm, avoiding stack overflow Y_F and X_F coordinates, shown in Eq. (1) can only be if:

$$P_F = YX . \quad (1)$$

This expression does not take into account the fact that the filling of the stacks of coordinates is accompanied by the selection of data from them, in consequence of which the inequality is always satisfied $P_F < YX$. Thus, it is necessary to accurately determine the size P_F FIFO-stacks Y_F and X_F coordinates of adjacent pixels when the image size $Y \times X$ pixels, presence of one segment on it ($N_s = 1$), the size of which $R_s(1)$, it matches with the image size ($R_s(1) = YX$).

3. Derivation an expression to accurately determine the size of the FIFO-stack of adjacent pixels coordinates

Let the size of the square image Y^2 pixels contains a single area of the same size. The work of the FIFO-stack in the process of segmentation of such an area can be divided into cycles. Within each cycle, from the FIFO-stack, all the coordinates of the pixels that were there at the beginning of the cycle are unloaded, performed their processing and loading the coordinates of pixels adjacent to the treated.

The number of coordinates are loaded into the FIFO-stack in each cycle depends on the position of the initial growth pixel (edge or center of the region). If the initial growth pixel is in the center of the square region, then in each cycle the number of elements in the FIFO stack increases by 8 (Fig. 3,a). Then the number of coordinates to be loaded into the FIFO-stack in the last $((Y-1)/2)$ - of cycle is $4(Y-1)$ on condition, Y - odd. If the initial growth pixel is in the corner of a square region, then in each cycle the number of coordinates in the FIFO-stack increases by 2 (Fig. 3,b). Then the number of coordinates to be loaded into the FIFO-stack in the last $(Y-1)$ - of cycle is $2Y-1$. Thus, when the region is segmented with the position of the initial growth pixel in its center, a larger capacity of stack is required (approximately 2 times) than in the case when the initial growth pixel is located in the corner of the region, and in an arbitrary location on the border of the region (Fig. 3, c).

To calculate the capacity of the stack, it is necessary to take into account the order of sampling the values of adjacent pixels. Surroundings corner of pixel (Fig. 4, a) may contain to 5-th unprocessed adjacent pixels, and the neighborhood of the other (non-corner) pixels - up to three pixels (Fig. 4, b). This means that when sampling the coordinates of a corner pixel from the FIFO stack, up to 5 coordinates of adjacent pixels may return to their place and the number of coordinates in the FIFO-stack will increase by 4, and when sampling the coordinates of a non-corner pixel from the FIFO-stack, up to three coordinates of adjacent pixels may return to their place and the number of coordinates in the FIFO-stack will increase by 2.

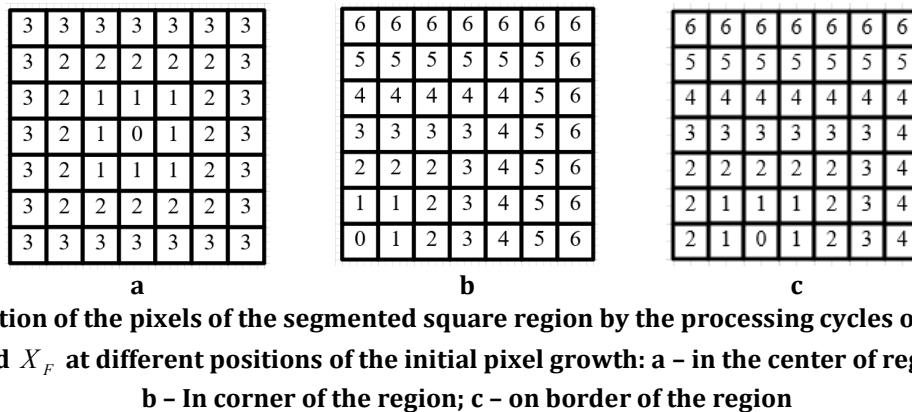


Fig. 3- Distribution of the pixels of the segmented square region by the processing cycles of FIFO-stacks Y_F and X_F at different positions of the initial pixel growth: a - in the center of region; b - In corner of the region; c - on border of the region

In this case, the segmentation of the square region with the initial growth pixel in the center of this region, the largest number of coordinates in the FIFO stack is formed. If during the processing of the neighborhood of the region, the adjacent pixels are selected not in order but with an interval of two pixels (Fig. 4, c) and sample begins with 4- angular pixels. Unloading from the FIFO-stack, the coordinates of 4 angular pixels of the grown region leads to the return to the FIFO-stack of coordinates of 16 adjacent pixels. Unloading from the FIFO stack, the coordinates of non- corner pixels with an interval of two pixels in k - cycle (total $4(\lfloor y/3 \rfloor - 1)$ of pixels when $y = 2k + 1$, $k = \overline{1, (Y-1)/2}$ and odd value Y), it leads to return to a FIFO-stack coordinates $(8(\lfloor y/3 \rfloor - 1))$ - of adjacent pixels, where $\lfloor \rfloor$ - rounding operation in downwards. Unloading from the FIFO-stack of the coordinates of the remaining not

angular pixels does not result in loading new coordinates in the stack (the coordinates of all adjacent pixels are already in the stack) and the number of coordinates in the FIFO-stack decreases.

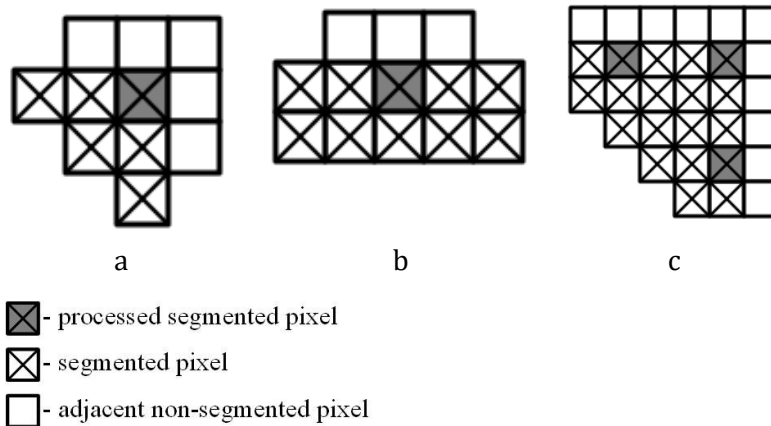


Fig. 4- Distribution of pixels of a segmented region:
a - in the neighborhood of the angular pixel; b - in the neighborhood of a non-angular pixel; c - when sampled with interval of two pixels

Thus, by the end of the cycle, the number of coordinates in the FIFO stack will increase by 8 compared with the previous cycle, but at the beginning of the cycle, the difference in the number of coordinates in the FIFO-stack between adjacent cycles can reach $16 + 8(\lfloor Y/3 \rfloor - 1)$ when sampling adjacent pixels with interval of two pixels, starting with corner pixels. In this case, in the last loop, when the size of the segmented area coincides with the image size, the number of coordinates in the FIFO-stack reaches the maximum value that determines the required size P_F of FIFO- stack (number of cells), calculated using an expression by Eq. (2):

$$P_F = 16 + 8(\lfloor Y/3 \rfloor - 1). \tag{2}$$

For more accurate expression, which allows to determine the required size P_F FIFO-stack in general, when the image size is $Y \times X$ pixels and $Y < X$ (Y, X - odd), It should be noted that the maximum load of stack is achieved by the segmentation of a rectangular region with an initial growth pixel in the center of this region into $((Y - 1)/2)$ -th cycle, when the size of the region is $Y \times Y$ (Fig. 5).

5	4	3	3	3	3	3	3	3	4	5
5	4	3	2	2	2	2	2	3	4	5
5	4	3	2	1	1	1	2	3	4	5
5	4	3	2	1	0	1	2	3	4	5
5	4	3	2	1	1	1	2	3	4	5
5	4	3	2	2	2	2	2	3	4	5
5	4	3	3	3	3	3	3	3	4	5

Fig. 5- The pixel distribution of the segmented rectangular area by the processing cycles of FIFO-stacks Y_F and X_F at the location of the initial of growth pixel in the center of the region

The following cycles occurs unloading of FIFO-stack coordinates with non-corner pixels of the interval of two pixels (by $2(\lfloor Y/3 \rfloor + 1)$ pixels in each cycle) and loading into FIFO-stack coordinates $(4(\lfloor Y/3 \rfloor + 1))$ - of adjacent pixels. Unloading from the FIFO-stack the coordinates of the remaining non-angled pixels does not load the new coordinates into the stack (the coordinates of all adjacent pixels are already on the stack) and the number of coordinates in the FIFO-stack decreases. Thus, the expression (2) can be used to determine the required size P_F FIFO- stack, when $Y < X$ and Y, X - an odd.

4. Evaluation of memory savings for the exact calculation of the size of FIFO-stack

In Fig. 6 shows the size dependence P_F of FIFO-stack (memory cells), defined approximately ($P_F^{(1)}$) using an expression (1) and accurately ($P_F^{(2)}$) using an expression (2), for the size YX of image (pixels).

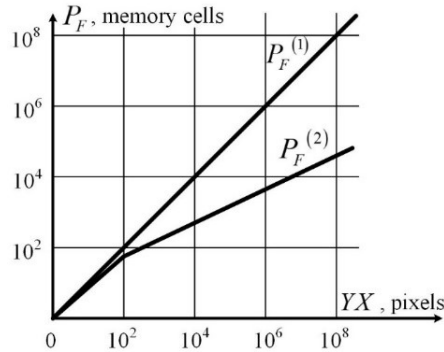


Fig. 6- Dependence of FIFO stack size on image size

The saving M_E of memory when accurate calculation of the required capacity for FIFO-stack by expression (2) in comparison with the approximate calculation of the expression (1) determined using the relation Eq. (3)

$$M_E = \frac{YX}{16 + 8(\lfloor Y/3 \rfloor - 1)} \quad (3)$$

Under the condition $Y < X$ and Y, X - an odd.

From the expression (3) and Fig. 6 it follows, that the memory savings for the exact calculation of the required capacity of the FIFO-stack increases with increasing image size and it is 36,8 and 374,3 times when the image size 100×100 and 1000×1000 pixels respectively.

5. Conclusion

An expression is obtained that allows to organize accurately the required size of the FIFO-stack in memory, required for storing the coordinates of adjacent pixels depending on the image size through a segmentation process based on region growing algorithm. The FIFO-stack, in memory organized on the idea of a ring multi-bit shift register. Upon obtaining of suggested expression, the condition of the maximum load of stack are accepted, when: a) a segmentation of a square region is performed with an initial of growth pixel in the center of this region.; b) during the processing for the surroundings of the region, adjacent pixels are not selected in order, but at interval of two pixels; c) sample begins with 4-th corner pixels. Using the proposed expression to calculate the required capacity of the FIFO-stack reduces the number of memory cells in 36,8 and 374,3 times when the image size 100×100 and 1000×1000 pixels respectively.

References

- [1] Z. Yiren, T. Do, H. Zheng, N. Cheung, and L. Fang, "Computation and memory efficient image segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, (2016), pp. 46-61.
- [2] A. Simmons, S. R. Arridge, P. S. Tofts, and G. J. Barker, "Application of the extremum stack to neurological mri," *IEEE Transactions on medical imaging*, vol. 17, (1998), pp. 371–382.
- [3] R.L. Kruse, "Data Structures and Program Design in C++," *New Jersey: Prentice-Hall*, (2000), 717 p.
- [4] S. Lipschutz and Schaum's, "Outline of Theory and Problems of Data Structures," *McGraw: Hill Book Company*, 1986, 344 pp.
- [5] M. Al-jawaherry and S. Hamid, "Image Compression Techniques: Literature Review", *JQCM*, (2021), vol. 13, no. 4, pp. 10.
- [6] N. Kaur, J. Singh and V. Sharma, "Analysis and comprehensive study: image segmentation techniques," *International Journal for Research in Applied Science & Engineering Technology*, vol. 3, pp. 241–246, 2015.
- [7] J. L. Moigne and J. C. Tilton, "Refining image segmentation by integration of edge and region data," *IEEE transactions on geoscience and remote sensing*, vol. 33, № 3, pp. 605–614, 1995.
- [8] O. Almiahi, and V. Kanapelka, "Progressive image segmentation based on the wave region growing," *Al-Sadeq International Conference on Multidisciplinary in IT and Communication Science and Applications*, (2016), pp. 1-6.
- [9] V. Dey, Y. Zhang, and M. Zhong, "A review on image segmentation techniques with remote sensing perspective," *IAPRS*. Vienna. Austria, vol. XXXVIII, (2010), pp. 31–42.
- [10] S. Hore, S. Chakraborty, S. Chatterjee, N. Dey, A. S. Ashour, L. V. Chung, G. Nguyen and D. Le, "An integrated interactive technique for image segmentation using stack based seeded region growing and thresholding," *International Journal of Electrical and Computer Engineering*, (2016), pp. 1–8.
- [11] S. Yuheng, Y. Hao, "Image segmentation algorithms overview," *ArXiv*, pp. 1-6, 2017.
- [12] Fasihi, M. Sadat, and W. B. Mikhael. "Overview of current biomedical image segmentation methods," *International Conference on Computational Science and Computational Intelligence (CSCI)*, (2016), pp. 803-808.
- [13] S. Kamdi, R. K. Krishna, "Image Segmentation and Region Growing Algorithm," *International Journal of Computer Technology and Electronics Engineering*, vol. 2, pp. 103–107, February 2012.
- [14] H. M. Hamed, O. M. Almiahi and O. Shauchuk, "Detection of anthropogenic objects based on the spatial characteristics of their contour in aerial image," *Indonesian Journal of Electrical Engineering and Computer Science*, (2021), pp. 206-215.
- [15] Ku.Vasundhara, H. Lokhande, "Study of region base segmentation method," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 4, pp. 197–200, 2014.
- [16] R. Brügger, C.F. Baumgartner, E. Konukoglu, "A Partially Reversible U-Net for Memory-Efficient Volumetric Image Segmentation," *Medical Image Computing and Computer Assisted Intervention*, (2019), 48 p.
- [17] A. Niemistö, T. Korpelainen, R. Saleem, O. Yli-Harja, J. Aitchison, and I. Shmulevich, "A k-means segmentation method for finding 2-d object areas based on 3-d image stacks obtained by confocal microscopy," *Proceedings of the 29th Annual International Conference of the IEEE EMBS, Lyon, France*, pp. 5559–5562, August 2007.
- [18] N. Wirth, *Algorithms and Data Structures*, N. Wirth: Prentice Hall, (1985), 212 pp.
- [19] S. Karmakar, M. Gogna, & F.C. Jain, "Application of quantum dot gate nonvolatile memory in image segmentation," *Signal, Image and Video Processing*, (2016), pp.551–558.
- [20] M. H. Hesamian, W. Jia, X. He and P. Kennedy, "Deep Learning Techniques for Medical Image Segmentation: Achievements and Challenges," *Journal of Digital Imaging*, vol 32, (2019), pp. 582–596.
- [21] T. Szirányi, J. Zerubia, L. Czúni, D. Geldreich, and Z. Kato, "Image Segmentation Using Markov Random Field Model in Fully Parallel Cellular Network Architectures," *Real-Time Imaging*, (2000), vol. 6, pp. 195-211.
- [22] Ch. Wuyang, J. Ziyu, W.Zhangyang, Kexin Cui, and Q.Xiaoning, "Collaborative Global-Local Networks for Memory-Efficient Segmentation of Ultra-High Resolution Images," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2019), pp. 8924-8933.
- [23] S. Vaze, W. Xie, and A. L. Namburete, "Low-Memory CNNs Enabling Real-Time Ultrasound Segmentation Towards Mobile Deployment," *IEEE Journal of Biomedical and Health Informatics*, (2020), vol. 24, pp. 1059-1069.
- [24] J. Sirotković, H. Dujmić and V. Papić, "K-means image segmentation on massively parallel GPU architecture," *Proceedings of the 35th International Convention MIPRO*, (2012), pp. 489-494.
- [25] A. Ali, M. RASHEED, S. SHIHAB, T. RASHID, A. Sabri, and S. Abed Hamed, "An Effective Color Image Detecting Method for Colorful and Physical Images", *Journal of Al-Qadisiyah for computer science and mathematics*, (2021), pp. 88-98.