Heliyon 10 (2024) e31152

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

# Image segmentation with Cellular Automata

Cesar Ascencio-Piña, Sonia García-De-Lira, Erik Cuevas *, Marco Pérez

*Departamento de Computación, Universidad de Guadalajara, CUCEI, Av. Revolución 1500, Guadalajara, Jal, Mexico*

ARTICLE INFO

*Keywords:*
Cellular automata
Image processing
Scientific computing
Noise reduction
Segmentation algorithms

ABSTRACT

Image segmentation is a computer vision technique that involves dividing an image into distinct and meaningful regions or segments. The objective was to partition the image into areas that share similar visual characteristics. Noise and undesirable artifacts introduce inconsistencies and irregularities in image data. These inconsistencies severely affect the ability of most segmentation algorithms to distinguish between true image features, leading to less reliable and lower-quality results. Cellular Automata (CA) is a computational concept that consists of a grid of cells, each of which can be in a finite number of states. These cells evolve over discrete time steps based on a set of predefined rules that dictate how a cell's state changes according to its own state and the states of its neighboring cells. In this paper, a new segmentation approach based on the CA model was introduced. The proposed approach consisted of three phases. In the initial two phases of the process, the primary objective was to eliminate noise and undesirable artifacts that can interfere with the identification of regions exhibiting similar visual characteristics. To achieve this, a set of rules is designed to modify the state value of each cell or pixel based on the states of its neighboring elements. In the third phase, each element is assigned a state that is chosen from a set of predefined states. These states directly represent the final segmentation values for the corresponding elements. The proposed method was evaluated using different images, considering important quality indices. The experimental results indicated that the proposed approach produces better-segmented images in terms of quality and robustness.

Nomenclature

| | |
|---|---|
| $m \times n$ | Dimension of the array |
| $x_{i,j}$ | A cell $x_{i,j}$ $(i \in 1,\ldots,m; j \in 1,\ldots,n)$ |
| $S$ | Set of states |
| $R_1,\ldots,R_r$ | Set of rules |
| $iter1$ | Iterations of phase 1 |
| $iter2$ | Iterations of phase 2 |
| $M_{3\times3}$ | Neighborhood of $3 \times 3$ |
| $x_0$ | Central element of a neighborhood |

## 1. Introduction

Image segmentation is an important field that has undergone significant advancements [1]. It is one of the most prominent techniques in image analysis, finding widespread applications in diverse domains, including medical imaging [2], military applications

* Corresponding author.
*E-mail address:* erik.cuevas@cucei.udg.mx (E. Cuevas).

https://doi.org/10.1016/j.heliyon.2024.e31152
Received 31 January 2024; Received in revised form 7 May 2024; Accepted 10 May 2024
Available online 11 May 2024

[3], and scientific research, where precise identification of specific image regions is crucial. This methodology involves partitioning an image into cohesive segments, each exhibiting homogeneous characteristics that influence the segmentation process. Similar to other computer vision methodologies, segmentation employs algorithms that rely on two key components: intensity value similarity and discontinuity [1,4–6].

Numerous segmentation techniques have been introduced in the literature that offer various approaches to tasks. These methods include Thresholding, Edge detection, machine learning-based, and metaheuristic algorithms. Thresholding is a widely used image segmentation technique that converts grayscale or color images into binary images by applying a predetermined threshold value [7]. Different strategies can be employed for threshold selection, including K-means clustering [8], the maximum entropy method, and Otsu's method based on maximizing the variance [9]. However, the efficacy of these strategies can vary depending on factors such as the noise levels in the image and the chosen threshold values, potentially leading to fuzzy segmentation outcomes [7,10].

Techniques based on edge detection represent another traditional approach to image segmentation with the aim of delineating regions with varying gray-level intensities by identifying edges within the image [11]. One well-known strategy in this realm is the Canny edge detection method, which detects potential edges based on a predefined threshold for the gradient magnitude. It utilizes four distinct filters on blurred images to identify diagonal, horizontal, and vertical edges. However, the presence of noise in images can significantly impact the accuracy of edge detection results, potentially leading to incomplete or irregular edge representations owing to misinterpretations of pixel continuity [12].

Machine learning-based image segmentation methods, as described by Kulwa et al. (2019) [4], have gained prominence in this field. In particular, deep learning (DL) techniques have yielded impressive results [13,14]. These approaches involve training neural networks on extensive datasets relevant to segmentation [15]. However, several challenges remain in the application of this method. First, effective interdisciplinary communication is not always guaranteed, making it difficult to obtain the requisite quantity and quality of the data. In addition, image noise and artifacts pose significant challenges in data preprocessing. Although this methodology is versatile and applicable in various domains involving computer vision, it is important to note that the nature of medical images can be highly diverse, presenting unique segmentation difficulties. Moreover, medical imaging datasets are often limited in their size and scope [15–17]. However, despite the interesting results of these approaches, they have critical drawbacks. These models require large amounts of labeled data for effective training. In many cases, obtaining such extensive and accurately annotated datasets is costly or impossible.

Metaheuristic algorithms [18] have emerged as viable approaches for image segmentation, particularly in the context of multilevel thresholding [19,20]. These algorithms have been extensively explored in the literature owing to their capacity to address complex segmentation tasks. However, it is worth noting that although they can yield promising results, some challenges persist. One significant limitation is the potential of metaheuristic algorithms to produce suboptimal solutions, which can lead to low-quality segmented images. This outcome is often attributed to issues related to the optimization performance, emphasizing the need for further research and refinement in their application to image segmentation tasks.

Most segmentation methods are highly sensitive to the presence of noise pixels generated during the image acquisition. In such scenarios, these approaches often exhibit poor performance and low accuracies. Noise, which can result from factors such as sensor imperfections or environmental interference, introduces inconsistencies and irregularities in image data. These inconsistencies affect the capacity of segmentation algorithms to distinguish true image features, leading to less reliable and lower-quality segmentation results.

Complex modeling refers to the use of advanced computational techniques to simulate and analyze systems that exhibit intricate, dynamic behaviors owing to the interaction of multiple factors and variables. These models are particularly valuable in fields such as heat and mass transfer [21,22], fluid modeling [23], biology [24], control [25], and social sciences, where the systems in question are too complex to be described by simple equations alone. Cellular Automata (CA) [26] is one of the most important paradigms of complex modeling that involves a grid of cells, each of which can be in one of a finite number of states. These cells evolve over discrete time steps based on a set of predefined rules that dictate how a cell's state changes according to its own state and the states of its neighboring cells. Cellular Automata are used to model and simulate complex systems and phenomena by observing how patterns and behaviors emerge from the interactions of simple individual cells. CA has applications in diverse fields, including physics, biology, and computer science. The main applications of CA include modeling physical phenomena, such as fluid flow and heat diffusion [27], simulating biological systems, such as cellular growth and population dynamics [28], and exploring emergent behaviors in complex systems, such as traffic flow or urban development [29]. CA's ability of CA to capture complex patterns and behaviors emerging from simple local interactions makes it a valuable tool for modeling, simulating, and analyzing dynamic systems in various scientific and computational domains. Although CA is an approach that uses a numerical two-dimensional array to characterize systems, its use in image processing is limited [30].

Noise and undesirable artifacts in images can cause significant inconsistencies and irregularities, which pose challenges for image-segmentation algorithms. These imperfections often hinder the ability of algorithms to accurately discern true features within images, resulting in less reliable and lower-quality segmentation outcomes. To address this issue, this paper presents a novel segmentation approach that leverages Cellular Automata (CA) models, which are adept at handling inconsistencies in image data. Unlike traditional segmentation methods, our proposed CA-based technique includes a three-phase process designed to robustly manage and mitigate noise and artifacts. The first two phases focus on reducing these disturbances by adapting the states of the cells or pixels based on their neighboring elements, thus improving the coherence and quality of the image data. In the final phase, each element is systematically assigned to one of the 16 predefined states that correspond to specific segmentation values. This structured approach ensures a clear and precise delineation of the image segments. Through rigorous testing of various images and evaluation using key quality indices such as PSNR, SSIM, and FSIM, our method demonstrated a superior capacity to produce high-quality, robustly segmented images. The

results underscore the effectiveness of our CA-based method in delivering enhanced segmentation performance, particularly under challenging conditions dominated by noise and artifacts.

The remainder of this paper is organized as follows. Section 2 discusses Cellular Automaton (CA) models. The proposed model is described in Section 3. Section 4 presents the experimental results and a comparative analysis. Finally, conclusions are presented in Section 5.

## 2. Cellular Automaton (CA)

Advanced computational techniques are employed in complex modeling to simulate and analyze systems with intricate, dynamic behaviors resulting from multiple factors and variables. This approach is particularly advantageous in various engineering disciplines [31,32] where the systems under investigation are too complex to be adequately described by simple equations.

Cellular Automaton (CA) is a complex modeling technique that consists of a grid-like structure, where each cell can be in one of a finite number of states. These cells were organized into rows and columns to forming a two-dimensional lattice. The evolution of CA occurs over discrete time steps, where each cell's state is updated based on a predefined set of rules that consider not only its own current state but also the states of its neighboring cells. This simultaneous updating of all cells according to the rules creates the dynamic behavior of the CA, allowing it to model and simulate complex processes and patterns through the interactions and transitions of individual cells in the grid.

### 1.1. Cells

In a Cellular Automata (CA) model, a cell serves as the fundamental building block of the system. In this paper, we restrict ourselves to two-dimensional spaces, which represent the most common cases. A two-dimensional model consists of a rectangular grid or lattice with $m \times n$ cells. Each cell $x_{i,j}$ ($i \in 1,\ldots,m; j \in 1,\ldots,n$) in the model has exactly one state at each time. For this, we define state set $= \{s_1, \ldots, s_q\}$. Therefore, the set consists of q states, and each agent should assume one of them.

### 1.2. Neighborhood

In Cellular Automata (CA) models, the neighborhood refers to the group of cells surrounding a given cell $x_{i,j}$ within the grid-like structure. The configuration of this neighborhood is defined by specifying which close cells are considered when determining the state transition of the central cell $x_{i,j}$. Two of the most common neighborhoods in Cellular Automata (CA) are the Moore neighborhood and the Von Neumann neighborhood.

In the Moore neighborhood, the central cell $x_{i,j}$ interacts with all eight neighboring cells surrounding it, including diagonally adjacent cells. This neighborhood is often used in CA models, which require extensive interactions between cells. In contrast to the Moore neighborhood, the Von Neumann neighborhood considers only the four orthogonal neighbors (north, south, east, and west) of a central cell. This neighborhood simplifies interactions, making them common in simpler CA models.

The size of a neighborhood can be extended to encompass a larger number of elements than the traditional 4 or 8. When a neighborhood involves more elements, it leads to a more substantial impact resulting from the interactions among them. In other words, a larger neighborhood size allows for a more comprehensive consideration of the spatial relationships and contextual information between cells in the CA grid. These neighborhoods dictate the scope of local interactions among cells, influencing how the state of each cell state changes based on the states of its neighbors.

### 1.3. Rules

A Cellular Automata (CA) model consist of a set of $r$ rules ($R_1, \ldots, R_r$). They specify how each cell changes its state based on its current state and the states of its neighboring cells. The precise formulation of these rules determines the behavior of the entire CA. Through these local interactions and state transitions, CA models can exhibit a wide range of complex and emergent behaviors. The rules essentially encode the underlying principles or mechanisms of the system being modeled, making CA an essential tool for simulating dynamic processes, studying pattern formation, and exploring the behavior of various systems, from physics simulations to biological growth and beyond.

## 2. Segmentation method by CA

The Cellular Automata (CA) model utilized in this study involved a grid with dimensions of $m \times n$ elements. Each cell $x_{i,j}$ within this array can assume one of the 256 states, which corresponds to different intensity levels in a grayscale image. At the beginning of the simulation, the initial state of the CA system was configured such that each cell was assigned an intensity level corresponding to pixel $p_{i,j}$ in the same location as the grayscale image $I(x,y)$.

The proposed approach consists of three phases. In the initial two phases of the process, the primary objective is to eliminate noise and undesirable artifacts that can interfere with the identification of regions exhibiting similar visual characteristics. To achieve this, a set of rules is employed in each iteration. These rules are designed to modify the state value of each cell based on the states of its neighboring elements. If the majority of a central cell's neighbors possess a higher intensity value, the state of the central element is

increased to promote the formation of a more homogeneous region. Conversely, if the majority of neighbors have a lower intensity level than the central cell, the state of the central cell decreases to encourage homogeneity within the region. In cases where there is no clear majority, half of the neighbors have higher values and half have lower values than the central element, the state of the central element remains unchanged, preserving its current state. This iterative process enhances the quality of image segmentation by promoting coherent regions while mitigating the impact of noise and irregularities.

The first two behaviors, which involve adjusting the state of the central element based on the majority of its neighbors, serve distinct purposes in the image segmentation process. In cases where the majority of neighbors have higher intensity values, the state of the central element is modified to promote homogeneity within the region. Conversely, when most neighbors have lower intensity levels, the state of the central element is adjusted to achieve a similar effect. In contrast, the third behavior, where there is no change in the state of the central element, plays a crucial role in highlighting the edges of the homogeneous regions.

These first two phases largely follow the same procedure, differing mainly in the size of the neighborhood considered to determine the majority in the rules. The initial phase employs a smaller neighborhood, whereas the second phase utilizes a larger neighborhood. These distinctions in neighborhood size led to varying effects resulting from the interactions between elements, allowing for the modeling of more intricate patterns and the production of homogeneities across the image. This multiphase approach enhances the segmentation process by both homogenizing regions and accentuating their boundaries, ultimately contributing to the accurate identification and delineation of distinct image features. Under these considerations, during the initial two phases, we can expect to encounter areas or objects characterized by their cohesive and easily discernible elements. This means that within these two stages, we will likely encounter sections or objects where the constituent components exhibit uniformity and can be readily distinguished from one another. The first and second stages are iterative phases that use the CA model. The first phase is applied during a set of *iter*1 iterations, whereas the second phase employs a set of *iter*2 iterations.

In the third phase of the segmentation process, each element was assigned a state chosen from a set of 16 predefined states. These states directly represent the final segmentation values for the corresponding elements. This phase essentially concludes the segmentation procedure by categorizing each element into one of 13 distinct segments, providing a clear and finalized segmentation outcome for the entire image. Fig. 1 illustrates the entire process of the proposed approach. As depicted in Fig. 1, the system workflow begins with an input image slated for segmentation, which may exhibit intensity variations stemming from diverse capture conditions. This image was processed using a two-stage approach. In the initial stage, a smaller neighborhood is employed to determine the operational rules. After a defined number of *iter*1 iterations, the outcomes were passed to the second stage. Here, processing considers a larger set of neighbors to facilitate pixel associations, and this stage continues for a designated number of *iter*2 iterations. Following the execution of these two stages, the image regions became homogeneous. In the third stage, a definitive value was assigned to each intensity value, signifying its ultimate state.

In the remainder of this section, each of the three phases used in our approach will be explained.

### 2.1. First stage

In the first stage, in the Cellular Automata (CA) model, each cell is evaluated within the context of an 8-neighborhood $\{x_1,\ldots,x_8\}$, equivalent to a $3 \times 3$ neighborhood arrangement as illustrated in Fig. 2.

Before establishing the specific rules that define the CA model, it becomes imperative to assess whether the neighboring elements possess higher or lower intensity levels in comparison to the central pixel $x_0$. To facilitate this assessment, a function denoted as $C(a,b)$ is introduced. This function serves the purpose of comparing the relative intensity levels between two arguments. This function can be

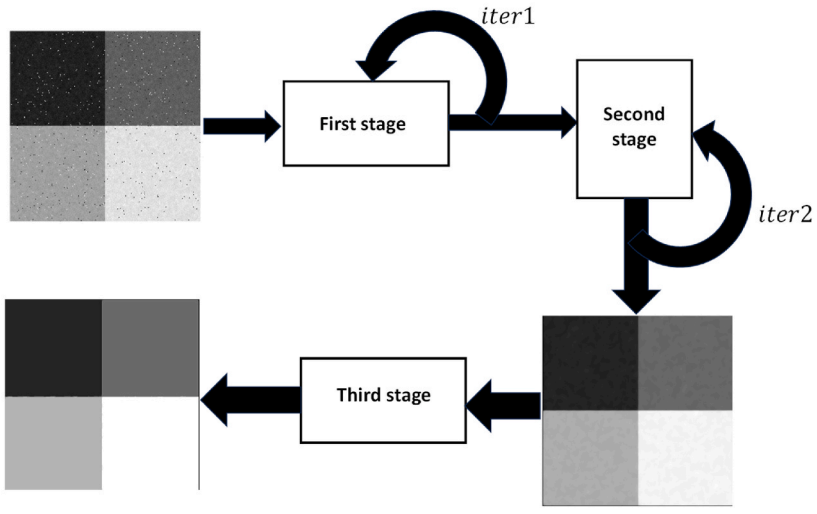


**Fig.1.** Illustration of the entire process of our approach.

| | | |
|---|---|---|
| $x_1$ | $x_2$ | $x_3$ |
| $x_4$ | $x_0$ | $x_5$ |
| $x_6$ | $x_7$ | $x_8$ |

**Fig. 2.** Employed neighborhood in the first stage.

modeled as follows:

$$C(a,b)=\begin{cases}1 & \text{if } a<b\\ -1 & \text{if } a>b\\ 0 & \text{if } a=b\end{cases} \tag{1}$$

Once function $C$ is introduced, the function majority of the central cell $M_{3\times3}(x_0)$ is defined. This function determines the behavior of the majority of neighbors with respect to the central element $x_0$. The structure is defined as follows:

$$M_{3\times3}(x_0)=\text{sign}\left(\sum_{i=1}^{8}C(x_0,x_i)\right) \tag{2}$$

In accordance with this function, when $M_{3\times3}(x_0)=1$, it signifies that the majority of the neighboring elements exhibit higher intensity levels in comparison to the central cell. Conversely, if $M_{3\times3}(x_0)=-1$, it indicates that the majority of neighboring elements possess lower intensity values than the central cell. Furthermore, when $M_{3\times3}(x_0)=0$, it carries a dual interpretation. First, it suggests that all elements within the neighborhood share the same state, meaning they have identical intensity levels. Second, it implies the absence of a clear majority, with half of the elements having higher intensity levels and the other half having lower levels. This scenario often means that $x_0$ is potentially an edge within the image, as there is no definitive majority in intensity levels among the neighboring pixels.

The CA model that we use at this stage considers three rules. These rules guide the state transitions of each cell within the array. Each rule associates a specific condition determined by the value of $M_{3\times3}(x_0)$ with the corresponding modification of the state for the central cell $x_0$. These rules play a central role in shaping the dynamic behavior of the CA model and are essential components in the process of image segmentation. The rules are outlined as follows:

$$\begin{aligned}R_1&: \text{If } M_{3\times3}(x_0)=1 \text{ then } x_0(k+1)=x_0(k)+1\\ R_2&: \text{If } M_{3\times3}(x_0)=-1 \text{ then } x_0(k+1)=x_0(k)-1\\ R_3&: \text{If } M_{3\times3}(x_0)=0 \text{ then } x_0(k+1)=x_0(k)\end{aligned} \tag{3}$$

$R_1$ is designed to address situations where the majority of the neighbors of the central cell exhibits higher intensity values. In such cases, the rule dictates that the value of $x_0$ should be incremented in the subsequent iteration $(k+1)$, aiming to promote homogeneity within the neighborhood. Conversely, $R_2$ is applied when the majority of neighbors surrounding $x_0$ have lower intensity values. In this scenario, the rule specifies that the value of $x_0$ should be decremented in the next iteration, aligning with the objective of homogenizing the neighborhood's intensity values. On the other hand, $R_3$ operates under two distinct scenarios. The first context arises when all elements within the neighborhood share the same intensity value, indicating a homogeneous region. In such cases, the value of $x_0$ remains unchanged to preserve the existing homogeneity. The second scenario is met when there is no clear majority within the neighborhood, with half of the neighbors having values greater than $x_0$ and the other half having values lower than $x_0$. In this case, $x_0$ is potentially located at an edge, and rule $R_3$ dictates that its value should not be modified in the subsequent iteration. This action aims to emphasize and highlight the edge rather than deleting it. These rules collectively guide the evolution of the CA model, contributing to effective image segmentation.

| | | | | |
|---|---|---|---|---|
| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
| $x_6$ | $x_7$ | $x_8$ | $x_9$ | $x_{10}$ |
| $x_{11}$ | $x_{12}$ | $x_0$ | $x_{13}$ | $x_{14}$ |
| $x_{15}$ | $x_{16}$ | $x_{17}$ | $x_{18}$ | $x_{19}$ |
| $x_{20}$ | $x_{21}$ | $x_{22}$ | $x_{23}$ | $x_{24}$ |

**Fig. 3.** Employed neighborhood in the second stage.

During phase 1, the CA model is executed during a predefined number of iterations *iter*1. The value of *iter*1 is 8.

### 2.2. Second stage

During the second stage of the process, every cell adopts its initial state as the final value obtained in the first stage. In this phase, the Cellular Automata (CA) model operates similarly to the first phase, utilizing the same fundamental principles and rules. However, a significant distinction lies in the size of the neighborhood considered around each cell. In this second stage, a larger and more extensive neighborhood configuration is employed, as depicted in Fig. 3. This enlarged neighborhood of $5 \times 5$ cells allows interactions on a broader scale, facilitating the modeling of more complex patterns and contributing to the formation of homogeneities across the image, further refining the segmentation process.

As the value of the neighborhood has been modified to a $5 \times 5$, the definition of the majority function $M_{5\times5}(x_0)$ is also redefined. Under these conditions, the function $M_{5\times5}(x_0)$ is modeled as follows:

$$M_{5\times5}(x_0) = sign\left(\sum_{i=1}^{24} C(x_0, x_i)\right) \tag{4}$$

In the second stage, the CA model utilizes the values of the majority function $M_{5\times5}(x_0)$ and applies the same set of rules as defined in Eq. (3), just as in the first stage. This iterative process continues for a specified maximum number of iterations, denoted as *iter*2. By applying the rules based on the majority of neighbors' intensity values in a larger neighborhood, the CA model refines and further enhances the segmentation, promoting homogeneity and pattern formation across the image. The value of *iter*2 is 8.

### 2.3. Third stage

After the completion of both the first and second stages, the elements within the Cellular Automata (CA) model undergo a process

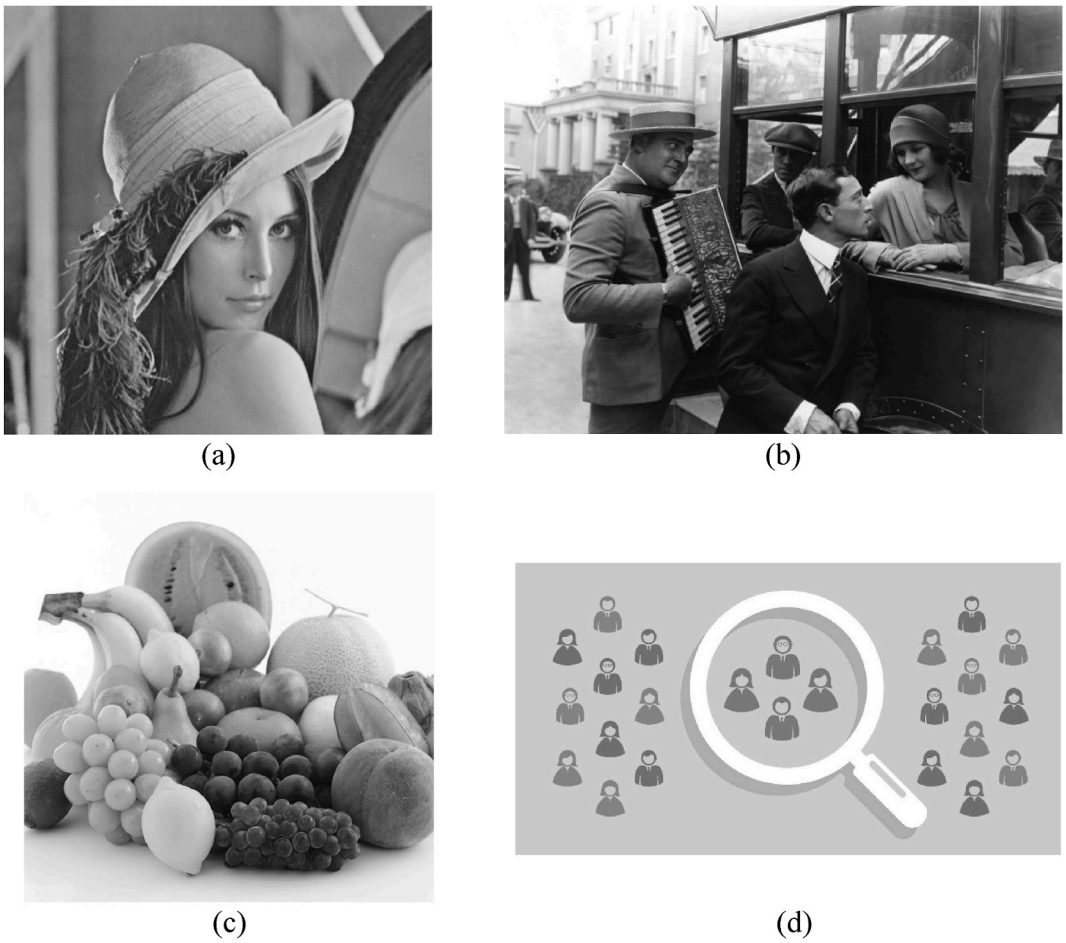


**Fig. 4.** Images used in the experiments.

that results in the identification of regions or objects characterized by homogeneous intensities and clearly defined edges. In the final stage of the segmentation process, each element is assigned a final state that represents the segmented object. This concluding step provides a definitive and interpretable outcome, as each element's state corresponds to a specific segment or object within the image, indicating the successful completion of the image segmentation procedure. To achieve final segmentation, the process involves dividing the range of intensity levels or states into a smaller number of segments. In this paper, the initial 256 intensity levels are condensed into 16 states. This reduction in the number of states is aimed at achieving two key objectives. First, it groups areas with similar intensity values into the same state, which potentially indicates that they belong to the same object or region. Second, it accentuates the differences between areas with dissimilar intensity levels and effectively highlights them as distinct and separate objects. This strategic reduction of states enhances the clarity and interpretability of the final segmentation, enabling a more effective representation of the objects and regions within the image.

In the third and final stage of the segmentation process, the cells within the Cellular Automata (CA) grid adopt as their initial state the final states that were reached during the second stage. In this phase, a specific rule $r_1$ is applied to these initial states, determining their transformation into the definitive segmented representation.

$$r_1 : \text{If } 0 \leq x_0(k) \leq 255 \text{ then } x_0(k+1) = round\left(\frac{x_0(k)}{16}\right) \tag{5}$$

where $round(\bullet)$ corresponds to the mathematical function that rounds a number to the nearest integer.

## 3. Experiments and results

In this section, we present a comprehensive analysis of the results obtained using the proposed approach. To assess its effectiveness and performance, we compared the outcomes with those of three well-established methods commonly found in the literature. These benchmark methods include DeepLab [13], which is a deep learning-based approach; the Firefly Algorithm (FA) [33], which represents metaheuristic optimization methods; and Otsu's method [34], a classical technique in the field of image segmentation. By conducting this comparative evaluation, we aim to provide a comprehensive perspective on the strengths and capabilities of the proposed approach in relation to these established methods, spanning the spectrum from classical to cutting-edge segmentation techniques.

To ensure a fair and unbiased comparison, the algorithms considered in our analysis were configured using the parameters recommended in their respective references. These specific parameter settings have been fine-tuned and validated to deliver optimal performance for each individual method.

To evaluate the efficacy of our segmentation methodology, we employed a comprehensive set of images from several public databases such as the Berkeley Segmentation Dataset and Benchmark (BSDS300) as our testing suites. In our analysis and comparison of the results, we considered a diverse set of images to comprehensively assess the performance of the proposed approach. However, owing to space constraints, we selected only four images known for their complexity. The results of the segmentation process are discussed in this section. These figures are shown in Fig. 4. The images presented in Fig. 4 illustrate the various challenges and characteristics utilized for testing the image processing algorithms. Fig. 4(a) features the iconic "Lena" image, renowned in the image processing community for its complexity, including a diverse array of objects and natural noise, making it a standard test image for algorithmic benchmarks. Fig. 4(b) displays the "People" image, which is notable for the contrast between objects of homogeneous intensity and those with intricate patterns featuring varying intensity levels, posing unique challenges in object differentiation. In Fig. 4(c), the "Fruits" image is shown, where contiguous objects share similar intensity levels, complicating the segmentation tasks as it is difficult to delineate one object from another. Finally, Fig. 4(d) introduces the "Magnifying Glass" image, where objects not only exhibit high-intensity levels but also possess closely matched values, thus complicating the task of distinguishing individual elements within the image. Each image was carefully selected to test different aspects of image processing techniques, highlighting the diverse challenges faced by algorithms in real-world applications.

All the experiments conducted for this study were executed using MATLAB 9.13 as the primary computational environment. The experiments were performed on a computer equipped with an Intel Core i7 processor running at 2.60 GHz, complemented by 32 GB of RAM and powered by an RTX 2070 graphics processor.

The experimental section of this study is structured into two distinct parts. In the first part (5.1), we conducted a visual comparison of the image results, focusing on their visual consistency and coherence. This qualitative analysis allowed us to assess the perceptual quality and overall effectiveness of each segmentation method in producing visually meaningful and consistent outcomes. In the second part (Section 5.2), we performed a quantitative assessment by comparing the numerical results. Specifically, we evaluated segmented images using commonly employed indices that gauge the quality and accuracy of the segmentation results. This quantitative analysis provides objective measures of performance, enabling a more precise evaluation of the performance of each method in terms of key image quality metrics. By combining both visual and numerical assessments, we aimed to present a comprehensive evaluation of the segmentation methods under consideration.

### 3.1. Visual comparison

In this section, we present a visual analysis of the results achieved by each of the algorithms under consideration. Our primary objective in this comparative examination is to gain insights into the abilities of the segmentation methods to effectively partition

objects within the images, particularly in the presence of noise or irregularities. The essence of this visual comparison lies in assessing whether these methods can produce coherent and meaningful divisions of objects, even in scenarios in which the images may contain noise or inconsistencies. To facilitate this analysis, Figs. 5–8 has been prepared, presenting the results generated by the four algorithms across the four images utilized in our experiments. This visual representation enables direct and comprehensive evaluation of the segmentation outcomes, shedding light on the performance of the methods in capturing and delineating distinct objects and regions within the images.

Fig. 5 illustrates the segmentation results for the "Lena" image, revealing the distinct effectiveness of various methodologies. Notably, the Cellular Automata (CA)-based method (Fig. 5(d)) emerges as the superior technique. This approach produces segments that are well-defined and closely align with the actual objects, highlighting its semantic precision. In contrast, the deep learning-based method (Fig. 5(a)) encounters difficulties, particularly in simultaneously handling all regions and addressing noise and artifacts, which detracts from its overall performance. Furthermore, the Firefly Algorithm (FA) (Fig. 5(b)) and the Otsu method (Fig. 5(c)) also display limitations in their segmentation capabilities. They struggle with the complexities introduced by unwanted pixel variations, leading to less satisfactory outcomes compared to the CA method. This visual assessment unequivocally accentuates the CA method's dominance in accurately segmenting the "Lena" image. The robust performance of the CA-based method can be attributed to its exceptional ability to homogenize segmented regions effectively. This capability ensures that the areas within the segmented image are not only consistent but also semantically coherent, which is crucial for accurately representing the objects depicted in the image. By maintaining a high level of internal consistency within these regions, the CA method excels at delineating objects with precision and fidelity. The inherent proficiency of this approach in generating coherent and semantically meaningful regions significantly enhances the quality of segmentation, enabling it to consistently outshine other methods. Ultimately, the algorithm's strength in region homogenization stands as a key contributor to its competitive edge in various image segmentation tasks.

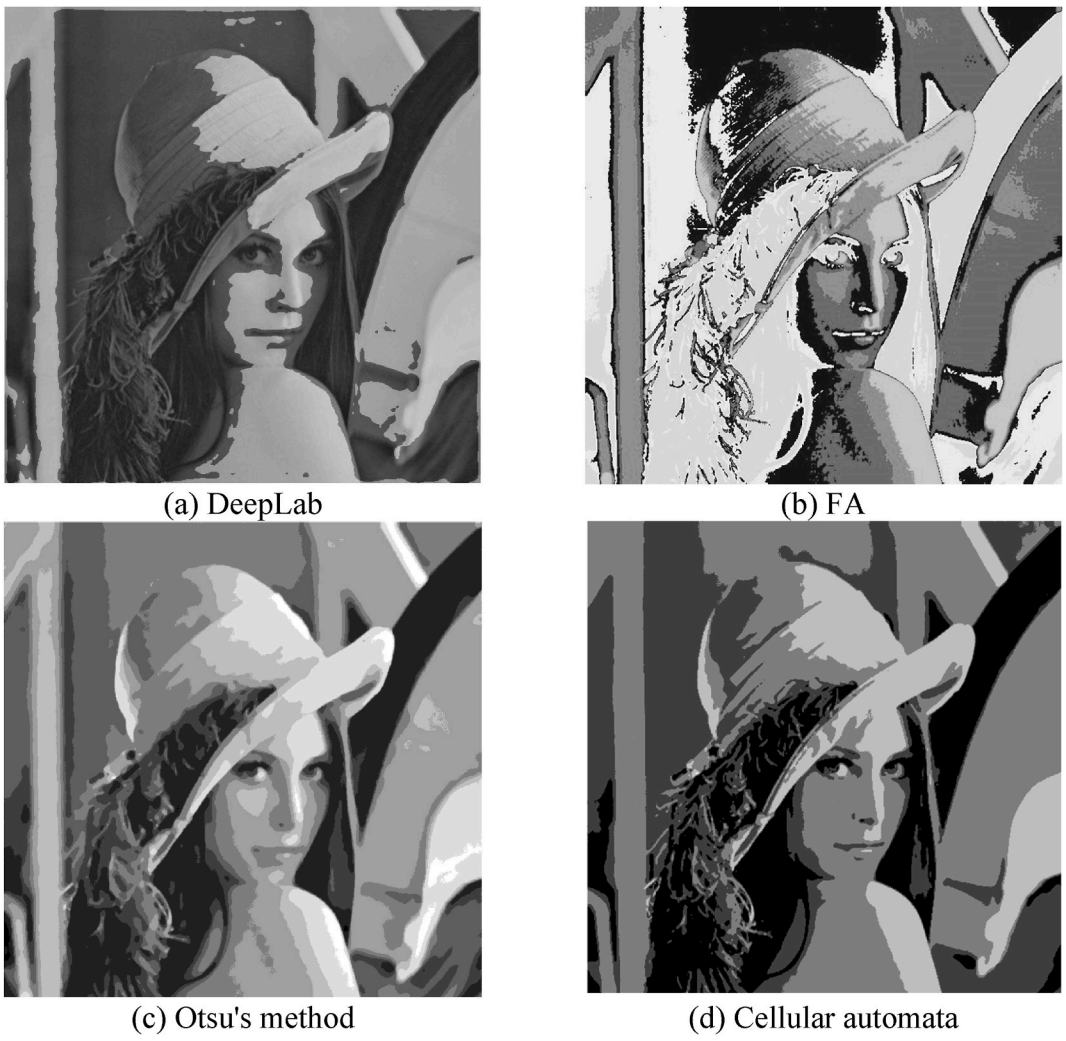


**Fig. 5.** Segmentation results of all methods considering the image "Lena", for the cases of (a) DeepLab, (b) FA, (c) Otsu's method and (d) Cellular automata.

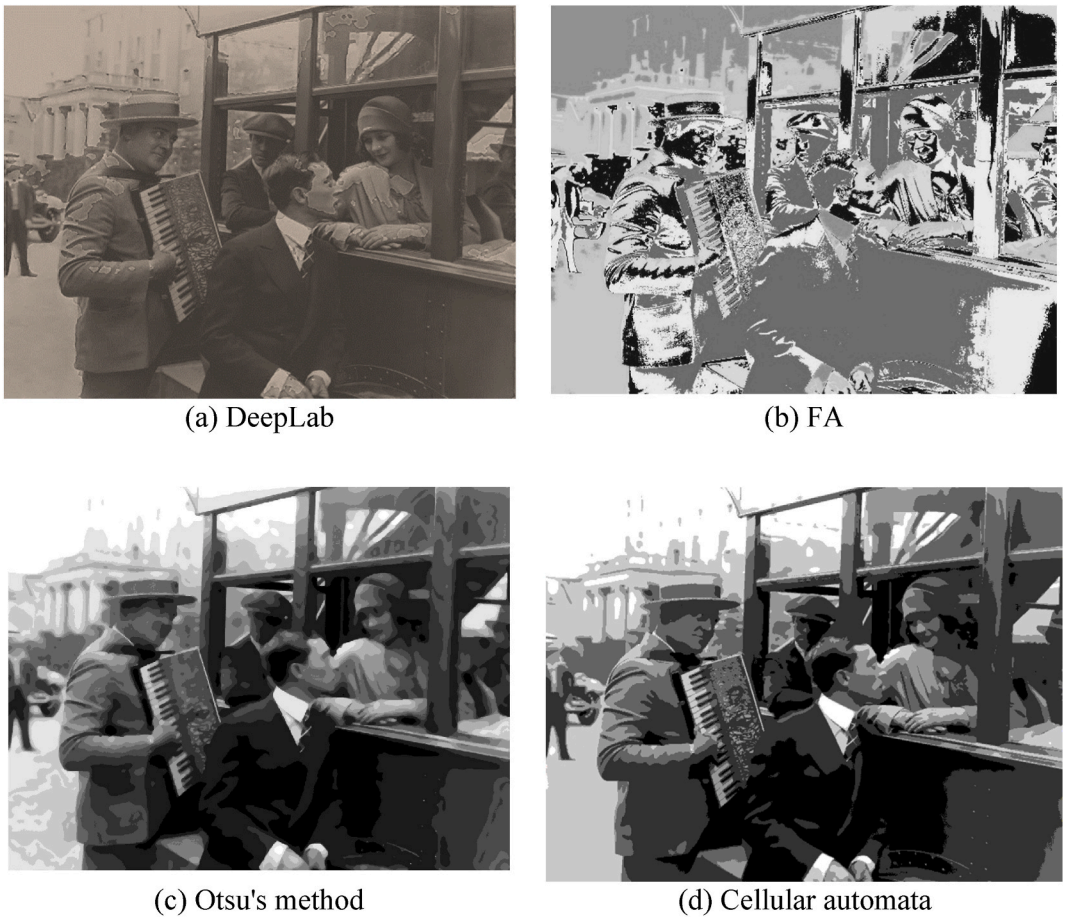


**Fig. 6.** Segmentation results of all methods considering the image “People”, for the cases of (a) DeepLab, (b) FA, (c) Otsu’s method and (d) Cellular automata.

Fig. 6 provides a comprehensive overview of the segmentation outcomes for the complex "People" image, characterized by intricate elements and a significant amount of noise and unwanted artifacts. Within this context, the Cellular Automata-based method (Fig. 6 (d)) once again emerged as the standout performer, yielding the most visually compelling results. This method exhibits a remarkable ability to effectively address unwanted variations present in the image while semantically partitioning certain objects, showcasing its versatility. In contrast, the deep learning-based method (Fig. 6(a)) struggles to achieve proper segmentation, possibly because the training images do not encompass such complex image contexts, leading to subpar performance. The results of the Firefly Algorithm (FA) (Fig. 6(d)) method indicate its limitations in handling strong intensity variations caused by varying capture conditions, resulting in a less effective segmentation. Finally, the Otsu method (Fig. 6(c)) tends to over segment regions and lacks the ability to semantically identify objects in an image. This comprehensive visual analysis underscores the superiority of the Cellular Automata-based approach in addressing the challenges posed by the "People" image.

Fig. 7 offers a detailed examination of the segmentation results for the "Fruits" image. Notably, the Cellular Automata-based method (Fig. 7(d)) stands out once again, delivering a more coherent and consistent segmentation that aligns well with actual objects within the scene. The segmentation regions generated by this method are semantically meaningful and offer visually pleasing outcomes. In stark contrast, the deep learning-based method (Fig. 7(a)) struggles to identify distinct regions primarily because it lacks the inherent knowledge necessary for comprehending the complex context of the image, resulting in suboptimal segmentation. The performance of the Firefly Algorithm (FA) method (Fig. 7(b)) visibly suffers in the presence of noise, leading to less effective segmentation. Finally, the Otsu method (Fig. 7(c)) tends to over segment objects, failing to tolerate small variations within regions that, in reality, belong to the same object. This analysis highlights the robustness and consistency of the Cellular Automata-based approach, particularly when dealing with challenging images like "Fruits".

Fig. 8 showcases the segmentation results for the intricate "Magnifying Glass" figure, which presents distinct challenges due to the difficulty in identifying individual elements and discerning the shadow cast by the magnifying glass on the background. A close examination of the figure reveals that the deep learning-based method (Fig. 8(a)) generally produces poor results and struggles to accurately segment the elements within the image. The FA-based method (Fig. 8(b)), on the other hand, faces limitations in identifying the diverse intensity levels present in the image, resulting in binary outcomes. In contrast, the Otsu method (Fig. 8(c)) demonstrated

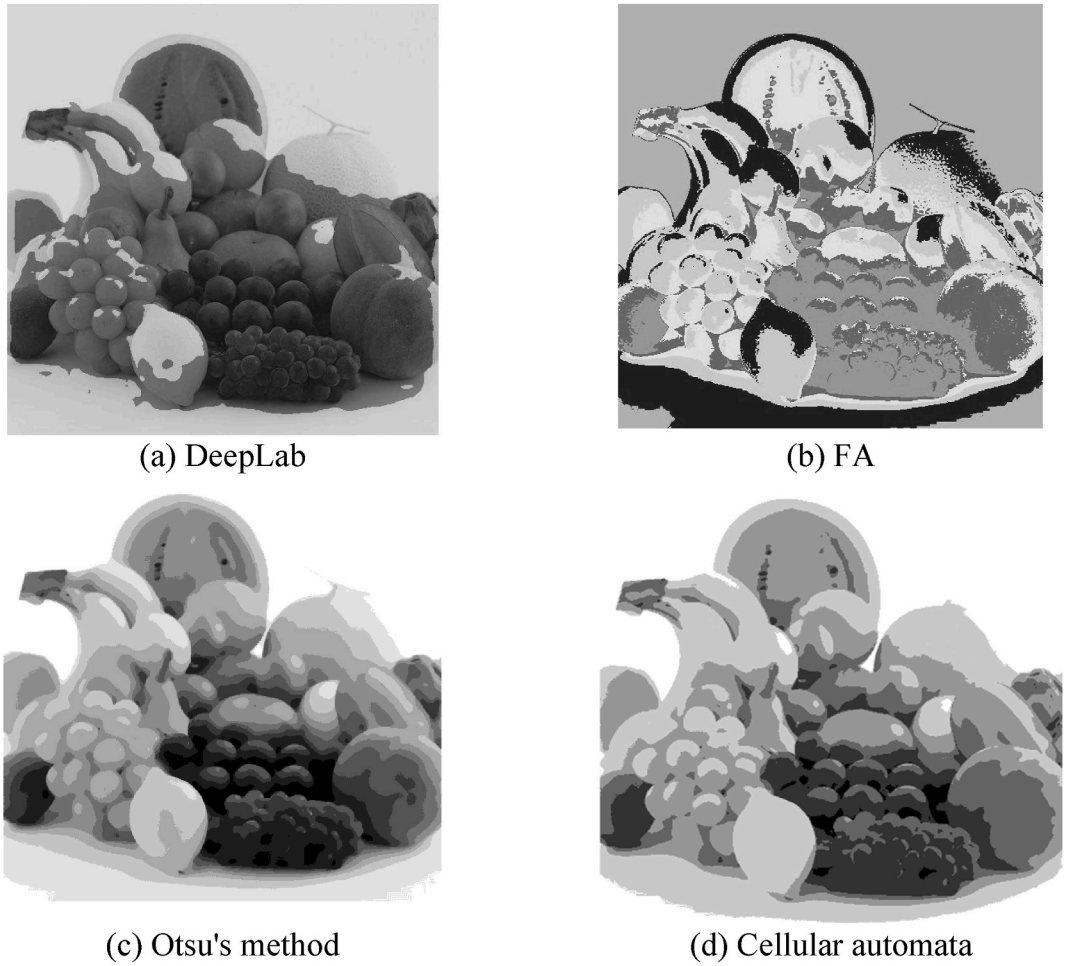


**Fig. 7.** Segmentation results of all methods considering the image "Fruits", for the cases of (a) DeepLab, (b) FA, (c) Otsu's method and (d) Cellular automata.

improved performance by identifying more objects, although it failed to identify all of them. The proposed method based on Cellular Automata (Fig. 8(d)) proved to be effective in identifying most objects within the image, with the exception of the shadow cast by the magnifying glass on the background. This analysis underscores the strengths of the Cellular Automata-based approach, particularly in handling complex segmentation tasks like those posed by the "Magnifying Glass" figure.

### *3.2. Numerical comparison*

To perform a numerical assessment of the image segmentation results, we employed a comprehensive set of established quality indices that offered an objective measure of performance. In this study, we utilized a range of metrics, including the Peak Signal-to-Noise Ratio (*PSNR*), the Structural Similarity Index Method (*SSIM*), and the Feature Similarity Index Method (*FSIM*). These specific indices were chosen for their proven ability to quantitatively and objectively evaluate the quality of the segmented images. By leveraging these metrics, we can precisely assess the accuracy and fidelity of the segmented outcomes, thereby establishing a robust and objective foundation for evaluating the performance of the segmentation algorithms.

The Peak Signal-to-Noise Ratio (*PSNR*) serves as a widely recognized metric for assessing the quality of a segmented image. It quantifies the extent of noise or distortion present in the segmented image by comparing it to the original unaltered image. A higher *PSNR* value signifies a lower degree of distortion or error, which indicates that the segmented image closely approximates the original image. Essentially, *PSNR* provides a quantitative measure of how effectively the segmentation process preserves the intricate details and essential information of the image. This metric is invaluable for evaluating the accuracy and fidelity of segmentation results, rendering it an indispensable tool in image analysis and processing. The *PSNR* was calculated as follows:

$$PSNR = 20\ log_{10}\left(\frac{255}{RMSE}\right) \tag{6}$$

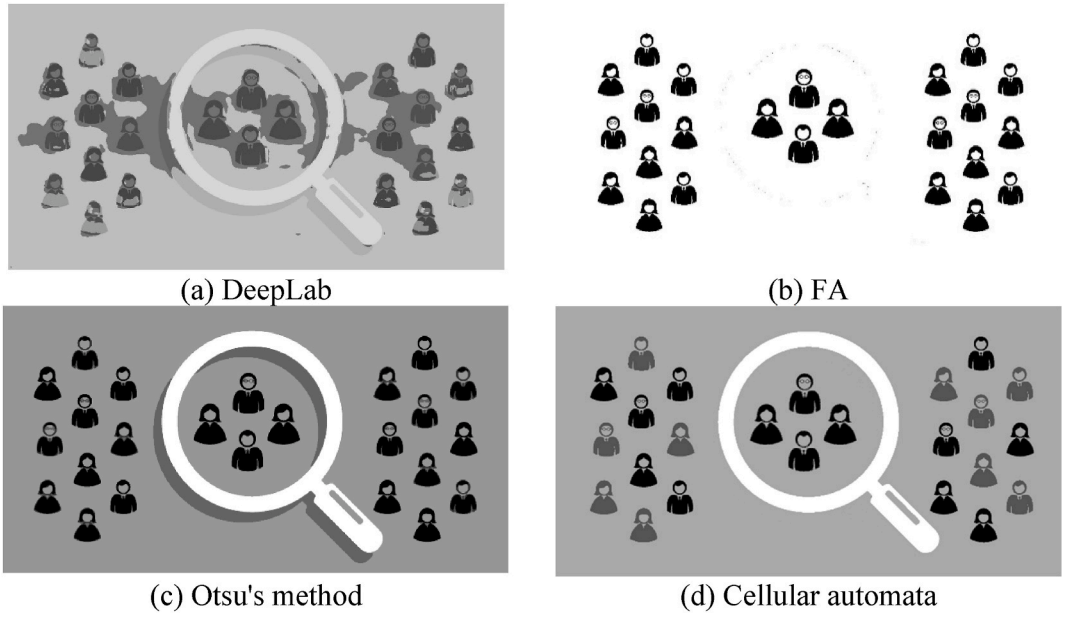


**Fig. 8.** Segmentation results of all methods considering the image "Magnifying glass", for the cases of (a) DeepLab, (b) FA, (c) Otsu's method and (d) Cellular automata.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{ro}\sum_{j=1}^{co}\left(I_{Gr}(i,j) - I_{th}(i,j)\right)}{ro \times co}} \tag{7}$$

The structural similarity index (*SSIM*) method is a robust metric employed to evaluate segmented images. SSIM gauges the structural similarity between the segmented image and a reference image, often the original, unsegmented image. It considers several key factors, including luminance, contrast, and structure, to assess how effectively the segmented image preserves crucial structural information inherent in the reference image. A higher *SSIM* score signifies closer correspondence between the segmented and reference images, signifying superior segmentation quality. *SSIM* offers a comprehensive evaluation of both local and global structural similarities, rendering it a valuable instrument for objectively appraising the accuracy and fidelity of segmented images. The *SSIM* was computed as follows:

$$SSIM\left(I_{Gr}, I_{th}\right) = \frac{\left(2\mu_{I_{Gr}}\mu_{I_{th}} + C1\right)\left(2\sigma_{I_{Gr}}\sigma_{I_{th}} + C2\right)}{\left(\mu_{I_{Gr}}^2 + \mu_{I_{th}}^2 + C1\right)\left(\sigma_{I_{Gr}}^2 + \sigma_{I_{th}}^2 + C2\right)} \tag{8}$$

$$\sigma_{I_0 I_{Gr}} = \frac{1}{N-1}\sum_{i=1}^{N}\left(I_{Gr_i} + \mu_{Gr}\right)\left(I_{th_i} + \mu_{I_{th}}\right) \tag{9}$$

where $\mu_{I_{Gr}}$ and $\mu_{I_{th}}$ are the mean value of the original and the segmented image respectively, for each image the values of $\sigma_{I_{Gr}}$ and $\sigma_{I_{th}}$ corresponds to the standard deviation. $C1$ and $C2$ are constants used to avoid the instability when $\mu_{I_{Gr}}^2 + \mu_{I_{th}}^2 \approx 0$ , experimentally in both values are $C1 = C2 = 0.065$..

The Feature Similarity Index Method (*FSIM*) served as a metric to assess the quality of the segmented images. FSIM's primary focus lies in the evaluation of structural and textural information within the segmented image, which is achieved by comparing it to a reference image, typically the original unaltered image. In contrast to conventional metrics that predominantly consider luminance and structural factors, *FSIM* places strong emphasis on capturing the perceptual quality of a segmented image. It scrutinizes crucial features, such as edges, textures, and patterns, aiming to determine how faithfully these essential visual elements are preserved during the segmentation process. A higher *FSIM* score denotes a closer resemblance between the segmented image and reference image, indicating superior segmentation quality concerning the preservation of vital visual features and textures. The *FSIM* is particularly valuable for assessing the perceptual quality of segmented images, rendering it an indispensable tool for diverse image analysis applications. The *FSIM* index was calculated as follows:

$$FSIM = \frac{\sum_{w\in\Omega} S_L(w) PC_m(w)}{\sum_{w\in\Omega} PC_m(w)} \tag{10}$$

where Ω denotes the domain of the image

$$S_L(w) = S_{PC}(w) S_G(w) \tag{11}$$

$$S_G(w) = \frac{2G_1(w)G_2(w) + T_2}{G_1^2(w) + G_2^2(w) + T_2} \tag{12}$$

$$G = \sqrt{G_x^2 + G_y^2} \tag{13}$$

$$PC(w) = \frac{E(w)}{\left(\varepsilon + \sum_n A_n(w)\right)} \tag{14}$$

Table 1 provides a detailed comparison of the numerical results derived from applying four different algorithms to the four images "Lena", "People", "Fruits" and "Magnifying glass". This analysis clearly shows that our proposed algorithm consistently outperforms the others across multiple evaluation metrics. Notably, in terms of Peak Signal-to-Noise Ratio (PSNR), our algorithm achieves a higher score, suggesting less distortion or error, thereby indicating a more accurate representation of the original image in the segmented version. Additionally, our method excels in the Structural Similarity Index Method (SSIM), which assesses the preservation of essential structural information that closely aligns with the reference image, highlighting its superior segmentation quality. Furthermore, our algorithm also leads in the Feature Similarity Index Method (FSIM), demonstrating an enhanced ability to maintain critical visual features and textures between the segmented and reference images, signifying its advanced segmentation capabilities. These strong performance indicators collectively underscore the efficacy and reliability of our proposed algorithm in handling image segmentation tasks. In comparison, the Deep Learning method ranks as the second-best, showing a commendable segmentation quality but not quite reaching the heights of our proposed method. The Otsu method occupies the third position, presenting reasonable results but still trailing behind the more sophisticated Deep Learning and proposed algorithms. Meanwhile, the Firefly Algorithm (FA) consistently performs the poorest, indicating significant shortcomings in comparison to the other methods. This comprehensive ranking not only highlights the superior performance of our proposed algorithm but also provides valuable insights into the strengths and weaknesses of each method in the context of image segmentation, thereby illustrating the distinct advantages of our approach.

The competitive results achieved by our proposed algorithm can be attributed to its remarkable ability to effectively homogenize segmented regions. This process consistently generates coherent areas within the segmented image, allowing for a faithful representation of the semantics of the objects present. By ensuring that the regions maintain a high degree of internal consistency, our algorithm excels at delineating objects with precision and fidelity. This inherent capacity to produce coherent and semantically meaningful regions plays a pivotal role in enhancing the segmentation quality of the algorithm, enabling it to consistently outperform alternative methods. In essence, the algorithm's prowess in region homogenization serves as a key factor contributing to its competitive performance across a range of image segmentation tasks.

## 4. Conclusions

This paper introduces a novel segmentation method that leverages Cellular Automata (CA) models, structured into three well-defined phases. The initial two phases focus primarily on the elimination of noise and undesirable artifacts, which can obscure the accurate identification of regions with similar visual characteristics. To achieve this, a specialized set of rules is implemented, which adjusts the state value of each cell or pixel according to the states of its neighboring elements. This adaptive rule application ensures that early processing steps effectively reduce image inconsistencies, paving the way for more accurate segmentation. In the third and final phase of the process, each element is assigned a specific state from a predefined set of possible states. These state assignments are not arbitrary but are carefully chosen to directly represent the final segmentation values of the respective elements. This structured approach ensures that each segment of the image is clearly defined and corresponds accurately to distinct visual traits, facilitating a high-quality segmentation output that is both reliable and practical for further image analysis tasks.

**Table 1**
Numerical indices produced by the segmented results.

| | | DeepLab | FA | Otsu | Cellular automata |
|---|---|---|---|---|---|
| **Lena** | *PSNR* | 17.8411 | 8.1746 | 8.845 | 22.702 |
| | *SSIM* | 0.7993 | 0.3959 | 0.4958 | 0.8821 |
| | *FSIM* | 0.8297 | 0.6844 | 0.7363 | 0.9710 |
| **People** | *PSNR* | 18.7835 | 8.1871 | 12.411 | 25.100 |
| | *SSIM* | 0.8256 | 0.2868 | 0.7521 | 0.9122 |
| | *FSIM* | 0.8624 | 0.6844 | 0.7500 | 0.9212 |
| **Fruits** | *PSNR* | 15.2200 | 8.5862 | 12.5107 | 21.129 |
| | *SSIM* | 0.8181 | 0.3952 | 0.7875 | 0.9314 |
| | *FSIM* | 0.8406 | 0.6875 | 0.8021 | 0.8609 |
| **Magnifying glass** | *PSNR* | 13.5778 | 11.002 | 13.3161 | 2403036 |
| | *SSIM* | 0.5898 | 0.2053 | 0.8418 | 0.9576 |
| | *FSIM* | 0.8581 | 0.8683 | 0.9327 | 0.9801 |

The proposed segmentation approach is simple, featuring just two adjustable parameters: *iter*1 and *iter*2. These parameters play a crucial role in modulating the intensity values of pixels, enabling them to adopt homogeneous values that correspond to the objects they represent, thereby effectively eliminating inconsistencies in the image. Both parameters are preset to a value of 8, though adjustments can be made depending on the specific requirements of the image being processed. If the value of *iter*1 and *iter*2 are lower, the removal of artifacts or unwanted pixels will be more subtle, which can be beneficial if the image features textures. However, if their values exceed 8, the process of merging distinct objects into one may become more aggressive.

To assess the efficiency and efficacy of our proposed segmentation method, we conducted a comparative analysis against three well-established segmentation methods frequently cited in the literature. These benchmark methods include DeepLab, which utilizes deep learning algorithms; the Firefly Algorithm (FA), a representative of metaheuristic optimization techniques; and Otsu's method, a classic threshold-based image segmentation technique. Our evaluation involved a comprehensive assessment of these methods across a diverse set of images, allowing for a thorough analysis of each method's performance in various scenarios. Due to constraints related to manuscript length and space, we focused our detailed analysis on four particularly complex images. These images were specifically chosen because their complexity provides a rigorous testing ground for demonstrating the robustness and adaptability of our proposed method compared to the established benchmarks. This selective focus ensures a clear, in-depth comparison of how each method handles challenging segmentation tasks, highlighting the advantages and potential limitations of our approach.

Our experimental analysis comprised two distinct segments. In the initial segment, we performed a visual examination of the images, emphasizing their visual uniformity and congruity. Subsequently, in the second segment, we conduct a quantitative evaluation by comparing the numerical outcomes. This evaluation involves the assessment of segmented images using widely accepted indices that measure the quality and precision of the segmentation outcomes.

An examination of the experimental data demonstrates that our proposed algorithm consistently outperforms other methods, as evidenced by multiple metrics and visual assessments. The competitive performance of our algorithm can be ascribed to its impressive capacity to standardize segmented regions effectively. This process systematically produces coherent areas within the segmented image, thereby ensuring an accurate representation of the semantics of underlying objects.

## Data availability statement

The images featured in this article are extracted from publicly accessible databases, each referenced for clarity and transparency.

## CRediT authorship contribution statement

**Cesar Ascencio-Piña:** Methodology, Formal analysis. **Sonia García-De-Lira:** Validation, Software. **Erik Cuevas:** Writing – original draft, Supervision, Investigation. **Marco Pérez:** Writing – review & editing, Visualization, Project administration.

## Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:Erik Cuevas is AE in Heliyon If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## References

[1] K.K.D. Ramesh, G. Kiran Kumar, K. Swapna, D. Datta, S. Suman Rajest, A review of medical image segmentation algorithms, EAI Endorsed Trans. Pervasive Heal. Technol. 7 (27) (2021) 1–9, https://doi.org/10.4108/eai.12-4-2021.169184.
[2] D.J. Withey, Z.J. Koles, A review of medical image segmentation : methods and available software, Methods 10 (3) (2008) 125–148.
[3] I. Segmentation, A Multi-Level Feature Fusion Network for Remote Sensing Image Segmentation, 2021.
[4] F. Kulwa, et al., A state-of-the-art survey for microorganism image segmentation methods and future potential, IEEE Access 7 (2019) 100243–100269, https://doi.org/10.1109/ACCESS.2019.2930111.
[5] Y. Zhang, Z. Shen, R. Jiao, Segment anything model for medical image segmentation: current applications and future directions, Comput. Biol. Med. 171 (2024) 108238, https://doi.org/10.1016/j.compbiomed.2024.108238.
[6] A. Basu, P. Senapati, M. Deb, R. Rai, K.G. Dhal, A survey on recent trends in deep learning for nucleus segmentation from histopathology images, Springer Berlin Heidelberg 15 (1) (2024), https://doi.org/10.1007/s12530-023-09491-3.
[7] W.X. Kang, Q.Q. Yang, R.P. Liang, The comparative research on image segmentation algorithms, in: Proc. 1st Int. Work. Educ. Technol. Comput. Sci. ETCS 2009, 2009, pp. 703–707, https://doi.org/10.1109/ETCS.2009.417, vol. 2.
[8] R.C. Gonzalez, Digital Image Processing, Pearson education india, 2009.
[9] R.M. Haralick, L.G. Shapiro, Image segmentation techniques, Comput. Vision, Graph. Image Process. 29 (1) (1985) 100–132, https://doi.org/10.1016/S0734-189X(85)90153-7.
[10] O.A. Arqub, Z. Abo-Hammour, Numerical solution of systems of second-order boundary value problems using continuous genetic algorithm, Inf. Sci. 279 (2014) 396–415, https://doi.org/10.1016/j.ins.2014.03.128.
[11] A. Thakur, R.S. Anand, A local statistics based region growing segmentation method for ultrasound medical images, Int. J. Medical, Heal. Pharm. Biomed. Eng. 1 (10) (2007) 565–570 [Online]. Available: http://www.waset.org/ijsp/v1/v1-3-26.pdf.
[12] N. Mahmood, A. Shah, A. Waqas, A. Abubakar, S. Kamran, S.B. Zaidi, Image segmentation methods and edge detection: an application to knee joint articular cartilage edge detection, J. Theor. Appl. Inf. Technol. 71 (1) (2015) 87–96.
[13] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT press, 2016.
[14] B. Panduri, O.S. Rao, A survey on brain tumour segmentation techniques in deep learning, Int. J. Intell. Syst. Appl. Eng. 12 (7s) (2024) 412–425.
[15] X. Liu, L. Song, S. Liu, Y. Zhang, A review of deep-learning-based medical image segmentation methods, Sustainability 13 (3) (2021) 1–29, https://doi.org/10.3390/su13031224.

[16] Q. Qin, Y. Chen, A review of retinal vessel segmentation for fundus image analysis, Eng. Appl. Artif. Intell. 128 (2024) 107454, https://doi.org/10.1016/j.engappai.2023.107454.
[17] J. Zhang, C. Ying, Z. Ye, D. Ma, B. Wang, Y. Cheng, Recent developments in segmentation of COVID-19 CT images using deep-learning: an overview of models, techniques and challenges, Biomed. Signal Process Control 91 (2024) 105970, https://doi.org/10.1016/j.bspc.2024.105970.
[18] O. Abu Arqub, Adaptation of reproducing kernel algorithm for solving fuzzy Fredholm–Volterra integrodifferential equations, Neural Comput. Appl. 28 (7) (2017) 1591–1610, https://doi.org/10.1007/s00521-015-2110-x.
[19] D. Oliva, M. Abd Elaziz, S. Hinojosa, Multilevel thresholding for image segmentation based on metaheuristic algorithms, in: Metaheuristic Algorithms for Image Segmentation: Theory and Applications, Springer International Publishing, Cham, 2019, pp. 59–69, https://doi.org/10.1007/978-3-030-12931-6_6.
[20] O. Zarate, D. Zaldívar, E. Cuevas, M. Perez, Enhancing pneumonia segmentation in lung radiographs: a jellyfish search optimizer approach, Mathematics 11 (20) (2023) 1–25, https://doi.org/10.3390/math11204363.
[21] S. Abbas, M. Nazar, Z.U. Nisa, M. Amjad, S.M.E. Din, A.M. Alanzi, Heat and mass transfer analysis of MHD jeffrey fluid over a vertical plate with CPC fractional derivative, Symmetry (Basel). 14 (12) (2022) 1–17, https://doi.org/10.3390/sym14122491.
[22] S. Abbas, M. Ahmad, M. Nazar, M. Amjad, H. Ali, A.Z. Jan, Heat and mass transfer through a vertical channel for the Brinkman fluid using Prabhakar fractional derivative, Appl. Therm. Eng. 232 (June) (2023) 121065, https://doi.org/10.1016/j.applthermaleng.2023.121065.
[23] S. Abbas, S.F.F. Gilani, M. Nazar, M. Fatima, M. Ahmad, Z.U. Nisa, Bio-convection flow of fractionalized second grade fluid through a vertical channel with Fourier's and Fick's laws, Mod. Phys. Lett. B 37 (23) (2023) 2350069, https://doi.org/10.1142/S0217984923500690.
[24] Z. Guo, et al., Diffusion models in bioinformatics and computational biology, Nat. Rev. Bioeng 2 (2) (2024) 136–154, https://doi.org/10.1038/s44222-023-00114-9.
[25] Y. Liang, et al., No title, Clin. Chem. Lab. Med. 62 (4) (2024) 635–645, https://doi.org/10.1515/cclm-2023-0964.
[26] S. Wolfram, Cellular automata as models of complexity, Nature 311 (5985) (1984) 419–424.
[27] L. Di Pietro, Strategies for describing preferential flow: the continuum approach and cellular-automaton fluids, in: Physical Nonequilibrium in Soils, CRC Press, 2022, pp. 437–453.
[28] C.A. Valentim, J.A. Rabi, S.A. David, Cellular-automaton model for tumor growth dynamics: virtualization of different scenarios, Comput. Biol. Med. 153 (2023) 106481.
[29] J. Lin, X. Li, Y. Wen, P. He, Modeling urban land-use changes using a landscape-driven patch-based cellular automaton (LP-CA), Cities 132 (2023) 103906.
[30] P.L. Rosin, Image processing using 3-state cellular automata, Comput. Vis. Image Underst. 114 (7) (2010) 790–802.
[31] S. Abbas, Z.U. Nisa, M. Nazar, M. Amjad, H. Ali, A.Z. Jan, Application of heat and mass transfer to convective flow of casson fluids in a microchannel with caputo–fabrizio derivative approach, Arabian J. Sci. Eng. 49 (1) (2024) 1275–1286, https://doi.org/10.1007/s13369-023-08351-1.
[32] S. Abbas, et al., Soret effect on MHD casson fluid over an accelerated plate with the help of constant proportional caputo fractional derivative, ACS Omega 9 (9) (2024) 10220–10232, https://doi.org/10.1021/acsomega.3c07311.
[33] K. Chen, et al., Multilevel image segmentation based on an improved firefly algorithm, Math. Probl Eng. 2016 (2016).
[34] P. Yang, W. Song, X. Zhao, R. Zheng, L. Qingge, An improved Otsu threshold segmentation algorithm, Int. J. Comput. Sci. Eng. 22 (1) (2020) 146–153, https://doi.org/10.1504/IJCSE.2020.107266.