

Article

Detection of COVID-19: A Metaheuristic-Optimized Maximally Stable Extremal Regions Approach

Víctor García-Gutiérrez, Adrián González , Erik Cuevas, Fernando Fausto and Marco Pérez-Cisneros 

Departamento de Ingeniería Electro-Fotónica, Universidad de Guadalajara (CUCEI), Blvd. Marcelino García Barragán #1421, Guadalajara 44430, Mexico; victor.ggutierrez@academicos.udg.mx (V.G.-G.); erik.cuevas@academicos.udg.mx (E.C.); abraham.fausto@academicos.udg.mx (F.F.); marco.perez@cucei.udg.mx (M.P.-C.)

* Correspondence: adrian.gonzalez@academicos.udg.mx

Abstract: The challenges associated with conventional methods of COVID-19 detection have prompted the exploration of alternative approaches, including the analysis of lung X-ray images. This paper introduces a novel algorithm designed to identify abnormalities in X-ray images indicative of COVID-19 by combining the maximally stable extremal regions (MSER) method with metaheuristic algorithms. The MSER method is efficient and effective under various adverse conditions, utilizing symmetry as a key property to detect regions despite changes in scaling or lighting. However, calibrating the MSER method is challenging. Our approach transforms this calibration into an optimization task, employing metaheuristic algorithms such as Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), Firefly (FF), and Genetic Algorithms (GA) to find the optimal parameters for MSER. By automating the calibration process through metaheuristic optimization, we overcome the primary disadvantage of the MSER method. This innovative combination enables precise detection of abnormal regions characteristic of COVID-19 without the need for extensive datasets of labeled training images, unlike deep learning methods. Our methodology was rigorously tested across multiple databases, and the detection quality was evaluated using various indices. The experimental results demonstrate the robust capability of our algorithm to support healthcare professionals in accurately detecting COVID-19, highlighting its significant potential and effectiveness as a practical and efficient alternative for medical diagnostics and precise image analysis.

Keywords: COVID-19 detection; X-ray image analysis; maximally stable extremal regions (MSER); metaheuristic optimization and image segmentation



Citation: García-Gutiérrez, V.; González, A.; Cuevas, E.; Fausto, F.; Pérez-Cisneros, M. Detection of COVID-19: A Metaheuristic-Optimized Maximally Stable Extremal Regions Approach. *Symmetry* **2024**, *16*, 870. <https://doi.org/10.3390/sym16070870>

Academic Editors: Lorentz Jäntschi and Jie Yang

Received: 29 April 2024

Revised: 20 June 2024

Accepted: 28 June 2024

Published: 9 July 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The COVID-19 pandemic has had a profound and multifaceted impact on societies across the globe, reshaping daily life, economic structures [1], and global health systems in unprecedented ways. In addition to the immediate health crisis, which has resulted in millions of infections and a significant number of deaths, the pandemic has triggered widespread social and economic disruptions. While necessary for public health, lockdowns and social distancing measures have led to job losses, business closures, and educational interruptions, exacerbating existing inequalities and presenting new mental health challenges [2]. The pandemic has had a far-reaching impact on society, with consequences that will be felt for years to come.

Classical COVID-19 diagnostics [3] rely on molecular tests, like RT-PCR, for early virus detection in individuals, and serological tests to identify past infections and assess immunity levels. While RT-PCR is crucial for quick isolation and treatment, serological testing helps us to understand vaccine effectiveness and population immunity. Despite their importance in managing the pandemic through informed public health decisions and clinical care, these methods face challenges such as limited testing capacity and the need for specialized equipment [4].

Considering the difficulties associated with conventional COVID-19 detection methods [5], the scientific community has explored alternative approaches, with the analysis of lung X-ray images emerging as a particularly promising option [6]. Despite its growing acceptance, the visual assessment of COVID-19 through radiographic images presents several challenges [7], primarily due to the subtle and highly variable radiographic features associated with the disease. The manifestations of COVID-19 in X-ray images often resemble other lung conditions, complicating the task of differentiating them based on visual examination [8]. This diagnostic challenge is further compounded by the presence of coexisting diseases, the overlap of symptoms among different conditions, and the variability in the expertise and experience of radiologists [9]. Such factors collectively contribute to the complexity of achieving an accurate diagnosis through radiographic visual analysis, emphasizing the need for enhanced diagnostic tools and methodologies.

The utilization of image segmentation methodologies represents a significant advancement in the visual analysis of lung X-ray images, particularly in assisting with the detection of COVID-19 [10]. These methodologies are crucial in enabling the precise delineation and isolation of areas affected by the virus, thus facilitating an accurate evaluation of disease magnitude and severity. Image segmentation provides essential quantitative insights that support clinical decisions, treatment strategizing, and the tracking of patient progress [11]. Such techniques significantly enhance the capability to distinguish COVID-19 and other pneumonia-related conditions from unaffected lung tissue, minimizing the likelihood of diagnostic errors. The incorporation of automated segmentation processes not only augments the efficiency and consistency of analyses but also delivers critical assistance to healthcare professionals [12]. This support is instrumental for the prompt and precise identification of pneumonia cases, contributing to superior patient management and outcomes.

The need for segmenting COVID-19-related anomalies in chest X-rays or radiographs has driven the adoption of image processing and computer vision techniques [13]. These techniques involve using thresholding to differentiate infected tissues from healthy ones based on pixel intensity variations, edge detection to define lesion boundaries, and region growing to group together pixels with similar attributes. Recently, researchers have employed machine learning methods, including convolutional neural networks (CNNs), to automate the segmentation process in pneumonia diagnosis [14]. CNNs are trained on labeled datasets to identify patterns and features indicative of pneumonia [15], allowing for accurate and efficient segmentation [16]. By integrating these approaches, researchers and healthcare professionals can achieve reliable and precise segmentation of COVID-19 abnormalities in radiographic images [17]. UNet [18] is a popular CNN architecture used primarily for image segmentation, particularly in the field of medical image analysis. Its original design consists of an encoder and a decoder with a contracting path and an expansive path. Variants of UNet have been created to enhance the original architecture and adapt it to different segmentation tasks. Some notable variants include UNet++, ResUNet [19], and DenseUNet [20]. Several methods based on CNN and deep learning techniques have also been proposed for the detection of pulmonary diseases, such as [21,22]. Although CNN and deep learning methods have been effective in segmenting COVID-19 abnormalities in X-rays or chest radiographs, they have some notable limitations. CNNs typically require a significant amount of labeled training data, which can be scarce and costly to obtain, especially for rare pneumonia cases [23,24]. Additionally, CNNs may struggle with generalization when applied to diverse patient populations or different imaging settings, potentially leading to inconsistent segmentation results.

Alternative methods for segmenting X-ray images for COVID-19 detection often utilize metaheuristic algorithms [25,26]. These algorithms turn the segmentation problem into an optimization challenge, enabling the efficient exploration of various configurations until the best results are found. To evaluate the segmentation outcomes, an objective function is constructed to compare different solutions. Many COVID-19 detection systems that employ metaheuristic techniques are combined with deep-learning neural networks [27]. In these instances, the optimization focuses on determining the neural network parameters that

yield the most effective segmentation results, with the objective function measuring the sum of errors between each training image and its segmented output. Various approaches using metaheuristic methods have been proposed in the literature, including Particle Swarm Optimization (PSO) [28], the Grey Wolf Optimizer (GWO) [29], the Firefly Algorithm (FF) [30], Genetic Algorithms (GA) [31], Cuckoo Search (CS) [32], Jellyfish Search (JS) [33], the Marine Predator Algorithm (MPA) [34], the Whale Optimization Algorithm (WOA) [35], artificial ecosystem optimization by means of fitness–distance balance (FDBAEO) [36], and the standard symbiotic organism search with the fitness–distance balance method (FDBSOS) [37]. Although these methods show promise, the process of collecting and labeling thousands of images with the help of human experts is time-consuming. Additionally, the limited availability of images often makes it challenging to gather a sufficiently large dataset for effective training.

Maximally stable extremal regions (MSER) [38] is a region detection method used in computer vision, designed to identify regions that remain consistent, or maximally stable, over a range of threshold values. These regions often correspond to meaningful objects or parts of objects within an image. MSER excels at detecting features in images where objects and the background have different intensities [39], making it challenging for other methods to identify them. In mathematical or physical contexts, symmetry refers to an object or system remaining unchanged under a specific set of operations. The MSER algorithm leverages this principle by detecting regions within images that maintain consistent properties despite variations in environmental conditions. This ability allows MSER to identify areas invariant to changes in scaling and lighting, enhancing its robustness in detecting significant features across various conditions. Such stability and invariance make MSER particularly effective in computer vision tasks [40] where consistent feature detection is crucial, regardless of external changes. Despite its effectiveness, configuring the MSER method's parameters to achieve optimal region identification can be challenging [41]. These characteristics have led to MSER being used in diverse applications, including line structure detection [42], anomaly identification [43], and region detection [44].

The challenge in applying MSER for image segmentation, however, lies in their various input parameters, which are usually indicated by the user (including the step size for the intensity thresholds to test, the maximum area variation allowed for the algorithm to label a region as stable, or the minimum/maximum number of pixels required for a region to be considered “relevant”). As one may expect, the selection of these parameters yields different detection results, which are hard to judge without some congruent criterion; this is what prompts us to interpret this parameter setup problem as an optimization problem, in which MSER parameters are treated as decision variables related to a minimization problem, which considers both the image entropy “ H ” and the cluster quality index “ C ” of output images produced as a result of a given parameter setup (as illustrated in Section 4.2). With this new insight into the problem, we are allowed to use a wide variety of optimization algorithms (including metaheuristic methods, which have shown great results over a wide range of problems) to find optimal configurations of MSER parameters for different image cases (in our case, lung X-ray images).

Thus, the novelty of our proposed approach may be summarized as follows:

1. The interpretation of the parameter selection problem in MSER-based image segmentation as an optimization problem (including the definition of an appropriate objective function);
2. Proposing metaheuristic algorithms for solving the previously mentioned optimization problem;
3. The application of both of the points mentioned above to lung X-ray image segmentation as a means to facilitate diagnosis (with a special interest in COVID-19).

The rest of this paper is organized as follows: Section 2 describes the maximally stable extremal regions (MSER) algorithm. Section 3 discusses the metaheuristic methods used in this study. Section 4 presents the proposed approach. Section 5 discusses the experimental

results. Section 6 discusses the complexity time and the accuracy in terms of the success rate. Finally, Section 7 presents the conclusions.

2. Maximally Stable Extremal Regions (MSER)

In this section, we explore the fundamental principles of the maximally stable extremal regions (MSER) algorithm [38], which forms the basis of our method for region detection. The MSER technique is based on a process similar to watershed segmentation, aimed at distinguishing regions of interest within a grayscale image. To formalize our discussion, let us consider a grayscale image I , defined by the function $I : \Omega \rightarrow [0, \dots, 255]$, where $\Omega = [1, \dots, M] \times [1, \dots, N]$ comprises all possible coordinates within the image. In the course of the MSER method, the intensity threshold t is adjusted from an initial value to a final value within the range of $[0, \dots, 255]$. This threshold serves to classify image pixels into two distinct groups: B (black), representing pixels with intensity values below the threshold, and W (white), representing those above it. By means of this binary classification, the MSER algorithm effectively pinpoints and isolates regions of the image that exhibit maximal stability across various threshold levels, thereby facilitating precise and accurate region detection.

As the threshold t is systematically adjusted from the minimum to the maximum intensity values, the composition of the two sets, B (black) and W (white), undergoes significant transformation. Initially, with t set at its minimum, all pixel positions are classified under W , rendering B empty, and the image appears entirely black. Upon gradually increasing t , isolated white spots begin to materialize within the black expanse, progressively expanding and coalescing as t further lowers. This dynamic expansion of white regions continues until, approaching the minimum threshold value, the once disjointed white areas unify into a single expanse, culminating in a fully white image—indicating that all pixels have transitioned to set W , leaving B devoid of any elements. Figure 1 illustrates this evolutionary process, showcasing the gradual changes in the image across various threshold levels, from dominantly black values to an entirely white one, marking the full spectrum of the thresholding effect.

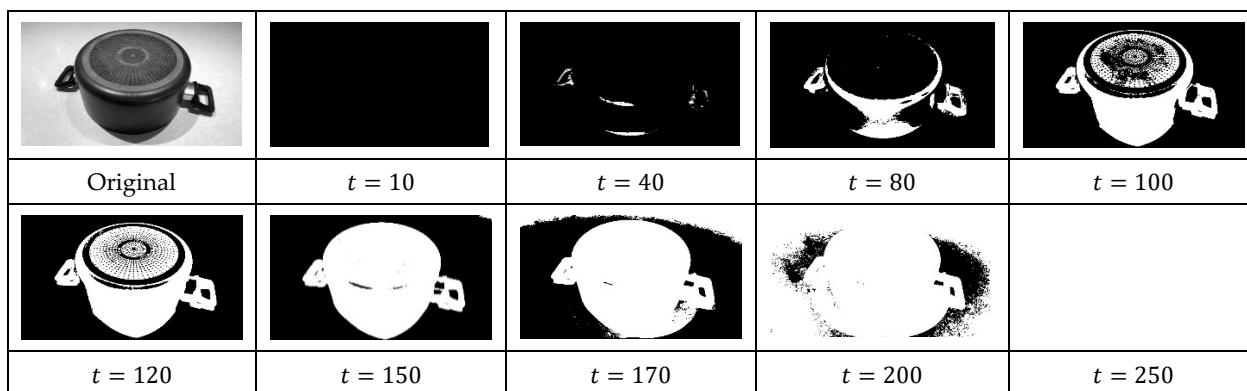


Figure 1. Evolutionary process, showcasing the gradual changes in the image across various threshold levels.

The primary objective of the algorithm [39] is to identify regions within the image that exhibit maximal stability over the broadest possible range of threshold intervals. This concept is visually elucidated through the representation of a 3D image in Figure 2a, where the focus is placed on a profile or line that demonstrates intensity variations across a singular dimension. As depicted in Figure 2b, the contour—integral to the region earmarked for detection—endures across multiple intensity levels ($t_0 - t_f$) before becoming fully enveloped. The aggregation of these levels, constituting the interval for this specific profile, exemplifies a region of exceptional stability. Essentially, these regions are characterized by their ability to maintain their structural integrity over a wide range of intensity thresh-

olds, signifying their significance and potential as stable features within the image. This approach underscores the algorithm's capacity to pinpoint areas of interest that are not only prominent but also remarkably consistent, making them invaluable for subsequent analysis or processing steps.

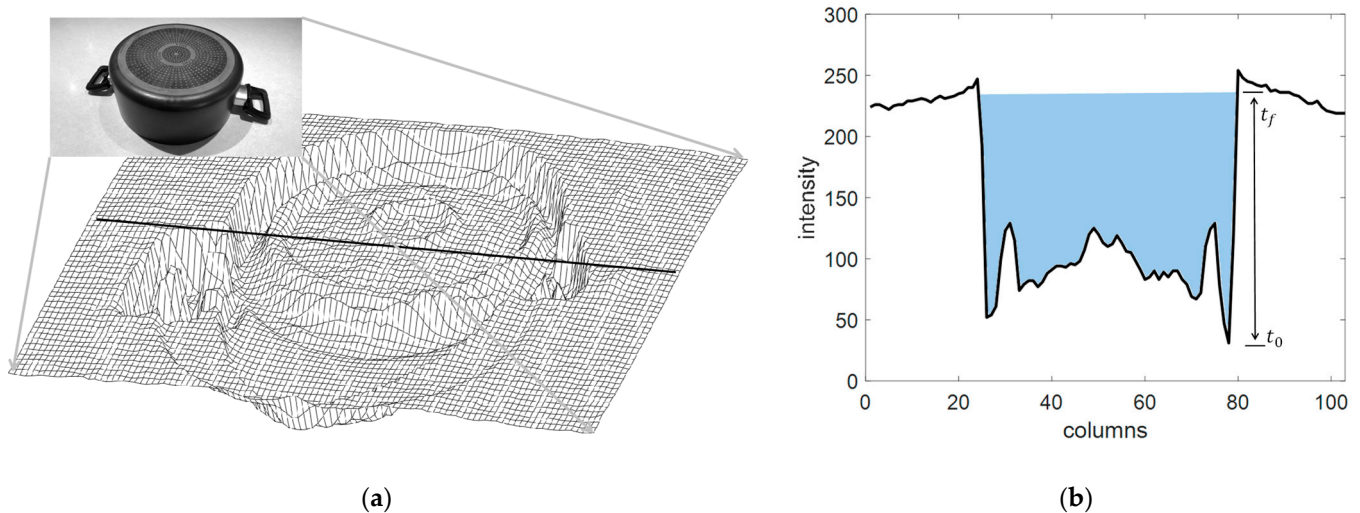


Figure 2. Process of varying the threshold values that define a region as stable. (a) 3D representation of the image and (b) interval of threshold values from which a profile is stable.

The core of the MSER method lies in evaluating the stability of these extremal regions as the threshold varies. A region is deemed maximally stable if its relative area changes minimally over a sequence of threshold levels, indicating a high level of robustness against changes in lighting, perspective, and other imaging conditions. This stability criterion is quantitatively assessed, with the algorithm selecting those regions that meet predefined stability thresholds over the widest possible range of intensity values. The output of the MSER algorithm is a set of regions that are candidates for further analysis or feature extraction, chosen for their distinctiveness and stability, making them highly suitable for tasks such as object recognition, image matching, and tracking.

Maximally stable extremal regions (MSER) are areas of an image, which only insignificantly vary under threshold changes. Formally, a binarized region $Q(t)$ is considered an extremal region if the growth rate function $q(t)$ produces a high value. The rate function is modeled as follows:

$$q(t) = \frac{\frac{d}{dt}(Q(t))}{\|Q(t)\|} \quad (1)$$

where $\frac{d}{dt}(\cdot)$ represents the derivative of the region area over the threshold value t , and $\|Q(t)\|$ symbolizes the number of pixels that the region $Q(t)$ involves.

The MSER algorithm operates with several parameters that control completely its behavior and performance. Adjusting these parameters allows for tuning the MSER detector to better suit specific applications or types of images. The main parameters typically include Delta (Δ), Minimum Area (*MinArea*), Maximum Area (*MaxArea*), Max Variation (*MaxVariation*), Min Diversity (*MinDiversity*), and Edge Blur Size (*EdBlu*).

Delta (Δ) determines the step size between intensity thresholds used to extract the extremal regions. A smaller delta value leads to more finely sampled thresholds, potentially detecting more regions but also increasing computational cost. *MinArea* specifies the minimum size of detected regions. This parameter helps in filtering out regions that are too small and that might be considered noise and not meaningful features. It is usually set as a percentage of the total image area. *MaxArea* defines the maximum size of detected regions, preventing the selection of overly large regions that may encompass significant portions of the image. Like the minimum area, this is often specified as a percentage of

the total image area. *MaxVariation* controls the variation in intensity within a region. It is a threshold for the stability criterion, where regions with a variation larger than this threshold are discarded. This parameter helps in distinguishing between stable regions and those that are too heterogeneous or not sufficiently cohesive. *MinDiversity* is used to filter out similar regions that are too close to each other in terms of their size. It ensures that the detected regions are diverse by eliminating those that are nearly identical, which helps in reducing redundancy among detected features. *EdBlu* determines the size of the kernel used for blurring the edges of the image before processing. Blurring can help in reducing the impact of noise and minor texture variations on the detection of extremal regions.

While the algorithm demonstrates good results, its efficacy is significantly influenced by the configuration of its parameters [45]. Achieving optimal results necessitates experimenting with a broad spectrum of parameter values, a process that can be both time-consuming and labor-intensive. This trial-and-error approach requires meticulous adjustment and testing of various combinations to identify the settings that best align with the desired performance criteria.

3. Metaheuristic Methods

A metaheuristic method [25] is a high-level problem-solving approach that applies heuristic strategies to explore and exploit the search space of complex optimization problems efficiently. These methods are designed to find good enough solutions to hard optimization problems within a reasonable timeframe, where finding the optimal solution might be impractical due to the problem's size or complexity or the computational resources required [46]. No single metaheuristic algorithm can outperform all others across all possible problems. This inherent limitation underlies the reason why a wide array of metaheuristic techniques exist, each tailored to leverage specific problem structures or to embody unique search strategies. Consequently, different methods exhibit varying degrees of efficacy depending on the nature and characteristics of the problem at hand. For this reason, it is important to identify the metaheuristic method that presents the best results depending on the problem to be optimized. Popular examples of metaheuristic methods include Particle Swarm Optimization (PSO) [47], the Firefly Method (FF) [48], the Grey Wolf Optimizer (GWO) [49], and Genetic Algorithms (GA) [50]. Each of these methods employs a different strategy to navigate the search space, balancing between the exploration of new regions and the exploitation of known good areas to avoid local minima and strive toward globally acceptable solutions. In the following, these metaheuristic methods are described, as they are used to identify the best method for detecting anomalies presented in an X-ray image by the MSER method.

3.1. Particle Swarm Optimization (PSO)

The Particle Swarm Optimization (PSO) [47] method is a metaheuristic technique inspired by the social behavior of birds flocking or fish schooling. It is used to solve optimization problems by simulating a swarm of particles (or agents) that explore the search space to find optimal solutions. In PSO, each particle represents a potential solution to the optimization problem and moves through the search space influenced by its own experience and that of neighboring particles.

The search strategy of PSO is governed by two main equations. The velocity update equation and the position update equation. These equations determine how particles adjust their trajectories based on their experiences and those of their neighbors. The velocity update equation v_i for each candidate solution i is modeled as follows:

$$v_i(k+1) = w \cdot v_i(k) + c_1 \cdot r_1 \cdot (p_{best,i} - x_i(k)) + c_2 \cdot r_2 \cdot (g_{best} - x_i(k)) \quad (2)$$

where $v_i(k+1)$ represents the velocity of particle i at iteration $k+1$, w is the inertia weight that controls the impact of the previous velocity on the current one, $v_i(k)$ is the velocity of particle i at iteration k , c_1 and c_2 are acceleration coefficients that control the influence of the personal best position ($p_{best,i}$) and the global best position (g_{best}) on the velocity update.

r_1 and r_2 are random numbers between 0 and 1, $x_i(k)$ is the current position of particle i at iteration k . On the other hand, the position update equation for the candidate solution i is computed as follows:

$$x_i(k+1) = x_i(k) + v_i(k+1) \quad (3)$$

This equation updates the position for the next iteration ($k+1$) based on its current position $x_i(k)$ and its newly calculated velocity $v_i(k+1)$. This step simulates the movement of the particle through the search space, allowing it to explore new areas for potential solutions.

3.2. Firefly Method (FF)

The Firefly Algorithm (FF) [48] is a metaheuristic optimization technique inspired by the flashing behavior of fireflies. The FF uses this behavior to move fireflies toward brighter (better) solutions, exploring the search space to find the global optimum. The search strategy of the FF Algorithm is primarily based on two key models: the attraction model and the movement equation.

The attractiveness of a firefly is determined by its solution quality (brightness), which decreases as the distance from other fireflies increases. The attractiveness β of a firefly can be modeled as follows:

$$\beta(r) = \beta_0 e^{(-\gamma r^2)} \quad (4)$$

where β_0 is the attractiveness at zero distance (i.e., its maximum value), γ is the light absorption coefficient that controls the decrease in light intensity, and r is the Euclidean distance between two solutions (fireflies).

The movement of a solution (firefly) toward another more attractive (brighter) firefly is governed by the following equation:

$$x_i(k+1) = x_i(k) + \beta(r)(x_j(k) - x_i(k)) + \alpha(rand - \frac{1}{2}) \quad (5)$$

where $x_i(k+1)$ is the position of the candidate solution i at time $k+1$, $x_i(k)$ and $x_j(k)$ are the positions of firefly i and another more attractive firefly j at time k , respectively, α is a parameter that adds a stochastic component to the movement, allowing exploration of the search space, $rand$ is a random number generated from a uniform distribution in the range $[0,1]$, and ensures that the random movement can be in any direction. The combination of these equations allows fireflies to explore the search space by moving toward brighter individuals, simulating the natural attraction behavior observed in fireflies. This movement strategy, coupled with the stochastic components, enables the Firefly Algorithm to balance exploration and exploitation effectively, making it capable of finding global optima across a wide range of optimization problems.

3.3. Grey Wolf Optimizer (GWO)

The Grey Wolf Optimizer (GWO) [49] is a metaheuristic optimization algorithm inspired by the social hierarchy and hunting behavior of grey wolves in the wild. Grey wolves are known for their leadership structure and cooperative strategy during hunting, which the GWO algorithm mimics to solve optimization problems. The algorithm models grey wolves' social hierarchy into four categories: Alpha (α), Beta (β), Delta (δ), and Omega (ω), where Alpha represents the best solution, followed by Beta and Delta as the second and third best solutions, respectively. The rest of the candidate solutions are considered Omega wolves.

Each wolf i updates its position x_i relative to the position of the prey x_p (the current best solution) using the following equations:

$$\begin{aligned} x_i(k+1) &= x_p(k) - A \cdot D \\ D &= |C \cdot x_p(k) - x_i(k)| \end{aligned} \quad (6)$$

where $x_i(k+1)$ is the position of the wolf i in the next iteration, $x_p(k)$ is the position of the prey, A and C are coefficient vectors, D is the distance between the wolf i and the prey p , and k is the current iteration.

3.4. Genetic Algorithms (GA)

Genetic Algorithms (GA) [50] are a class of evolutionary algorithms that mimic the process of natural selection and genetics to solve optimization and search problems. Each potential solution, often referred to as an individual, is encoded typically as a string of bits, characters, or numbers. The fundamental operations that involve the search strategy of GA are selection, crossover, and mutation.

Selection is an operation that mimics the survival of the fittest principle from natural selection. Individuals are selected based on their fitness (how well they solve the problem at hand) to reproduce and pass their genes to the next generation.

Crossover is an operation used to combine the genetic information of two parents to generate new offspring. This process introduces diversity into the population and allows for the creation of potentially better solutions.

Mutation introduces random changes to individual genes in an offspring to maintain genetic diversity within the population. It helps prevent the algorithm from becoming too homogeneous and getting stuck in local optima.

3.5. Artificial Ecosystem Optimization by Means of Fitness–Distance Balance (FDBAEO)

The artificial ecosystem optimization by means of the fitness–distance balance (FDBAEO) [36] model is a nature-inspired optimization algorithm tailored for engineering design optimization. It emulates the interactions and dynamics of natural ecosystems to explore and exploit the search space efficiently. In FDBAEO, candidate solutions, emulating species in an ecosystem, undergo processes such as reproduction, competition, and cooperation. The key aspect of this model is the balance between fitness (how good a solution is) and distance (how different a solution is from others), which helps in maintaining diversity and avoiding local optima. This balance is crucial for the algorithm's global search capability and robustness. The most important equations in FDBAEO include the fitness evaluation function, $f(x)$, which quantifies the quality of a solution x , and the fitness–distance balance equation defined as follows:

$$FDB_i = \alpha \cdot f(x_i) + \beta \cdot D(x_i) \quad (7)$$

where α and β are weighting factors, and $D(x_i)$ represents the distance of solution x_i from others in the population. These equations drive the selection and evolution processes, ensuring that the population evolves toward optimal solutions while maintaining diversity.

3.6. The Standard Symbiotic Organism Search with the Fitness–Distance Balance Method (FDBSOS)

The FDB-based symbiotic organism search (FDBSOS) [37] is an advanced optimization technique that enhances the standard symbiotic organism search algorithm by integrating the fitness–distance balance (FDB) method. This hybrid approach aims to improve solution diversity and avoid premature convergence by balancing the fitness of solutions with their distance from one another. In the context of SOS, organisms (candidate solutions) interact through mutualism, commensalism, and parasitism phases, mimicking natural symbiotic relationships to evolve towards optimal solutions. The FDB method enhances this process by ensuring that selected organisms are both high-quality and diverse. To ensure that only the best individuals participate in the process, various selection mechanisms are used to select the best elements. These operations not only allow to obtain the best individuals but also promote diversity, thus avoiding premature convergence.

4. The Proposed Method

In this section, the methodology and computational framework of our proposed approach are described in detail, focusing on the utilization of the maximally stable extremal

regions (MSER) method for the identification of anomalous regions indicative of COVID-19. The effectiveness of the MSER technique in delineating these characteristic regions is significantly influenced by the precise configuration of its parameters, which are intrinsically image-specific. Given this dependency, the task of region detection via MSER evolves into an optimization challenge, with the primary objective being the identification of an optimal set of MSER parameters. These parameters are essential for enhancing the algorithm's capability to accurately identify relevant regions within the image. To achieve this, an objective function is utilized, designed to quantitatively assess the quality and representativeness of the detected regions, thereby guiding the optimization process toward the most effective parameter configuration for the MSER algorithm. This approach ensures a systematic and data-driven mechanism to maximize the algorithm's efficiency in detecting regions potentially associated with the COVID-19 disease.

4.1. Problem Statement

In the proposed approach, the parameters of the MSER algorithm, once initialized, are iteratively modified by a metaheuristic optimization algorithm until the optimal value for the identification of abnormalities attributed to COVID-19 disease in X-ray images is reached.

The MSER algorithm operates with six parameters that completely control its behavior and performance. These parameters involve the Delta (Δ, x_1), Minimum Area ($MinArea, x_2$), Maximum Area ($MaxArea, x_3$), Max Variation ($MaxVariation, x_4$), Min Diversity ($MinDiversity, x_5$), and Edge Blur Size ($EdBlu, x_6$). Under these conditions, the decision variables are the six that will be represented by the vector $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5, x_6)$. To optimize these parameters, it is important to explore the entire configuration space. To perform this exploration, it is necessary to know the bounds within which the search space is constrained.

The minimal value of Δ can be as low as one, indicating that the algorithm will evaluate every possible threshold level. Its upper limit is generally set based on practical considerations to avoid processing unnecessary threshold levels. A common upper limit might be around 25, as higher values might skip meaningful thresholds and reduce the sensitivity of the feature detection.

The values of the parameters $MinArea$ and $MaxArea$ are expressed as percentages of the total image area. $MinArea$ could be as low as 0% (in practice, a small positive percentage avoids detecting noise as a feature). The maximal value of $MinArea$ is often not more than 1–5% of the total image area to avoid filtering out too many potential features. The minimal value of $MaxArea$ is set slightly above $MinArea$ to ensure a range of area sizes are considered. Often, it might start at 1–5% of the image area. The maximal value of $MaxArea$ is set to a value within 50% of the image area to prevent the selection of overly large, non-discriminative regions.

If the minimal value of $MaxVariation$ is close to 0, it indicates a very strict requirement for region stability. In practice, a typical upper limit of $MaxVariation$ might be around 0.5, as higher values would permit too much variation within a region, reducing the specificity of the feature detection.

The minimal value of $MinDiversity$ is 0, indicating no requirement for diversity among detected regions (not commonly used, as it defeats the purpose of the parameter). Its maximal value can be up to one, with values closer to one enforcing greater diversity between the selected regions.

The $EdBlu$ represents the size of the blur kernel, which is typically chosen based on the resolution of the image and the scale of features of interest. The minimal value $EdBlu$ of is a kernel of 3×3 , whereas its maximal value is 9×9 .

Therefore, the objective of the proposed method is to identify the global solution for a nonlinear problem based on the formulation of an optimization problem, described as follows:

$$\begin{aligned} &\text{Minimize } J(\mathbf{x}) \quad \mathbf{x} = (x_1, x_2, x_3, x_4, x_5, x_6) \in \mathbb{R}^6 \\ &\text{Subject to } \mathbf{x} \in X \end{aligned} \quad (8)$$

where $J : \mathbb{R}^6 \rightarrow \mathbb{R}$ is a six-dimensional nonlinear function, and X represents the search space defined by $1 \leq x_1 \leq 25$, $0 \leq x_2 \leq 5$, $1 \leq x_3 \leq 50$, $0 \leq x_4 \leq 0.5$, $0 \leq x_5 \leq 1$ and $x_6 \in \{3, 5, 7, 9\}$.

4.2. Objective Function

The objective function plays a pivotal role, as it quantitatively evaluates the quality of potential solutions, serving as a guiding metric for the optimization process. The objective function assigns a value to each possible solution, reflecting how well it meets the optimization criteria. This valuation enables the differentiation between more and less desirable solutions, guiding the search toward the optimal solution by systematically comparing the outcomes of various candidate solutions. In optimization algorithms, particularly in metaheuristic methods, the objective function is crucial for directing the exploration and exploitation phases, ensuring that the algorithm's adjustments to solution parameters are consistently aimed at improving the function's value.

To measure the quality of the regions segmented by the MSER method generated by each set of parameters, an objective function J is defined that measures the quality of this segmentation. The objective function combines the value of the entropy H and the quality of the clustering of the segmented regions C .

The entropy H is used as a measure of the information content or the variability of pixel intensities within a region [51]. A correct division of regions typically features homogenous regions, where the pixel values are very similar, leading to lower entropy values. Conversely, poorly segmented regions, characterized by a mix of various pixel values, exhibit higher entropy due to the increased randomness and complexity in the distribution of pixel intensities. The entropy of an image is evaluated using an expression that quantifies the unpredictability or randomness of the pixel intensity levels. For region detection, the entropy H of a region can be calculated using the following expression:

$$H = -\sum_{i=0}^{L-1} p(i) \cdot \log_2(p(i)) \quad (9)$$

where L is the number of intensity levels in the image (for an 8-bit grayscale image $L = 256$), $p(i)$ represents the normalized histogram or the probability of occurrence of intensity level i within the identified region, $\log_2$ denotes the logarithm to base 2, aligning with the information theory concept of entropy measured in bits.

The quality of the clustering C [42] is a metric that is important in evaluating the quality of image segmentation or the detected regions, as segmentation can be viewed as a form of clustering where pixels are grouped based on their similarities. This metric assesses how well the segmentation algorithm groups similar pixels together and separates dissimilar ones across different segments. To measure the quality of clustering in our approach, we will use the following expression:

$$C = \sum_{i=1}^k \sum_{x \in R_i} \|x - \mu_i\|^2 - \alpha \sum_{i=1}^{k-1} \sum_{j=i+1}^k \|\mu_i - \mu_j\|^2 \quad (10)$$

where $\{R_1, \dots, R_k\}$ denotes the set of regions identified by the MSER algorithm, k represents the total number of regions, and R_i contains the pixels within region i . Here, x refers to an image pixel, and μ_i is the centroid of region i . The notation $\|\cdot\|$ symbolizes the Euclidean distance, and α , a regularization parameter, adjusts the trade-off between intra-cluster compactness and inter-cluster separation (with a typical value of 0.6). In Equation (1), the objective of the first term is to minimize the total squared distances between each data point and the centroid of its respective cluster, thereby promoting highly compact clusters. Conversely, the second term, which is subtracted from the first, aims to maximize the distances between the centroids of different clusters, thus ensuring that each cluster is distinctly separated from the others. Therefore, the lower the value of C , the better the quality of the segmentation.

Once the entropy H and clustering quality C are defined, the objective function J used in our approach combines both elements as follows:

$$J = \min(H, C) \quad (11)$$

The combination of entropy and clustering quality metrics provides a dual perspective on segmentation quality. They allow for a balanced evaluation that not only focuses on the consistency of pixel intensities within each segment but also on the overall effectiveness of the segmentation in dividing the image into meaningful, well-defined parts.

4.3. Computational Procedure

In our proposed approach, the MSER method is used to detect anomalies in an X-ray image caused by COVID-19. For their correct detection, the optimal parameters of the algorithm are obtained through an optimization process. In order to identify the best metaheuristic optimization algorithm to solve this problem, four different methods have been proposed that use Particle Swarm Optimization (MSER-PSO), the Firefly Method (MSER-FF), the Grey Wolf Optimizer (MSER-GWO), Genetic Algorithms (MSER-GA), artificial ecosystem optimization and fitness–distance balance (MSER-FDBAEO), and the standard symbiotic organism search with the fitness–distance balance method (MSER-FDBSOS).

Typically, our methodology begins with the generation of an initial population of N candidate solutions $\{x_1, \dots, x_N\}$. Each solution x_i represents a set of parameters $x_i = (x_{1,i}, \dots, x_{6,i})$ that configure the MSER method. These candidate solutions are then iteratively improved through various mechanisms depending on the used metaheuristic technique (PSO, FF, GWO, GA, FDBAEO, or FDB-SOS). Throughout the process, metaheuristics employ strategies to avoid getting trapped in local optima and to ensure a comprehensive search of the space. This is often achieved through adaptive mechanisms that adjust the search dynamics based on feedback from the current search state. Each iteration of the algorithm evaluates candidate solutions using the objective function J that quantifies how close a parameter set is to the optimal result. The iterative process continues until a stopping criterion is met, which could be a set number of iterations, a convergence threshold, or other problem-specific conditions. Figure 3 shows a representation of the process performed by the proposed methodology.

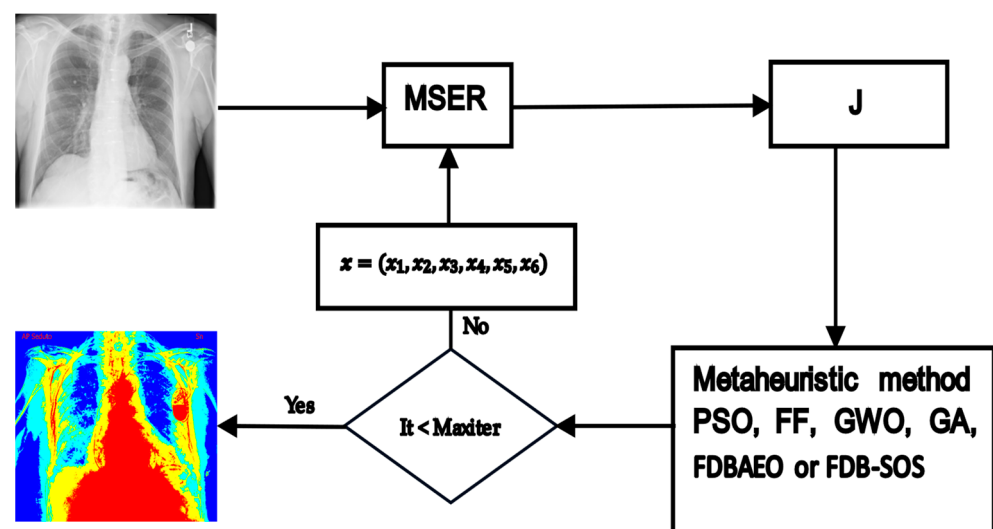


Figure 3. Representation of the process performed by the proposed methodology.

5. Experimental Results

In this section, we explore the experiments designed to evaluate the effectiveness of our methodology in detecting COVID-19 anomalies in X-ray images. These experiments

involved a detailed analysis of visual characteristics and performance metrics using a dataset of various lung radiographs. We performed a comparative study to assess the efficacy of our approach. Due to space limitations, we present results from a subset of the dataset (extracted from <https://www.kaggle.com> accessed on 12 September 2022), specifically five digital lung images: three from healthy individuals and two from COVID-19 patients. Figure 4 shows the images that have been selected. In Figure 4, images 1–3 are images of healthy individuals and images 4 and 5 are images corresponding to individuals with COVID-19, which contain abnormalities.

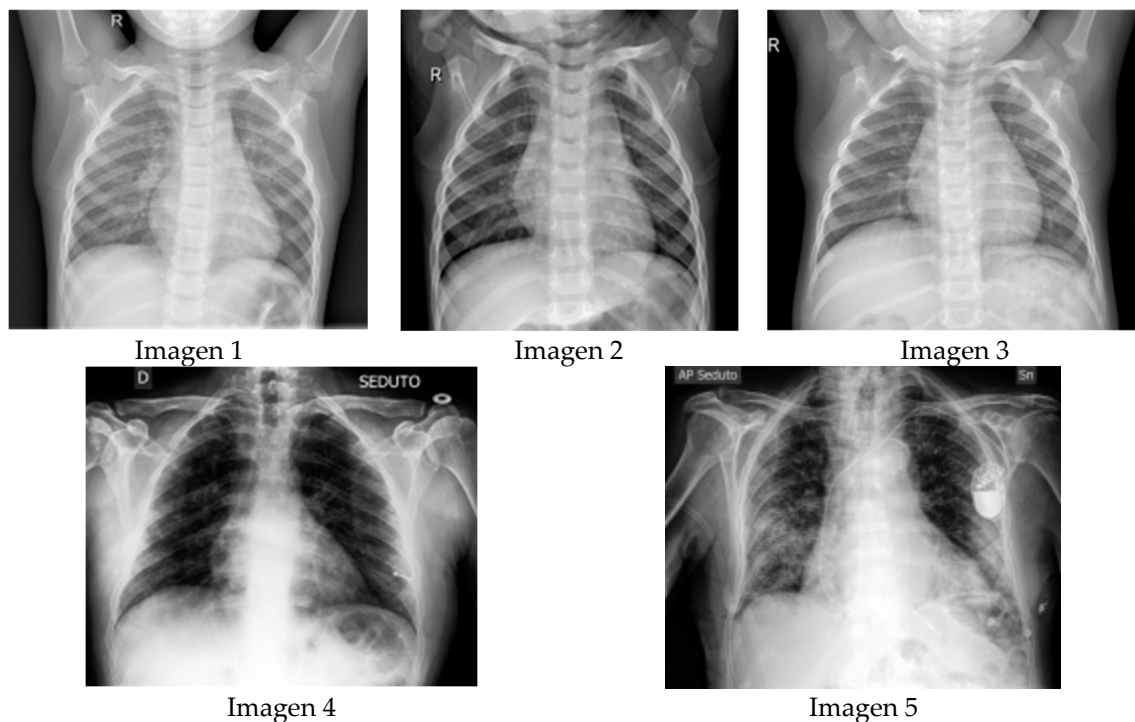


Figure 4. Images used in the computational experiments; images 1–3 are images of healthy individuals and images 4 and 5 are images corresponding to individuals with COVID-19.

In our experiments, we explored the effectiveness of combining the MSER method with four distinct metaheuristic algorithms—Particle Swarm Optimization (PSO) [47], Firefly (FF) [48], the Grey Wolf Optimizer (GWO) [49], Genetic Algorithms (GA) [50], artificial ecosystem optimization and fitness–distance balance (FDBAEO) [36], and the standard symbiotic organism search with the fitness–distance balance method (FDBSOS) [37]. This combination resulted in the creation of four detectors, MSER-PSO, MSER-FF, MSER-GWO, MSER-GA, MSER-FDBAEO, and MSER-FDBSOS. For comparative analysis, each metaheuristic method was configured with specific parameters, as detailed in Table 1. These parameter settings were selected based on recommendations from existing literature, where they are cited as enabling the best performance of the respective methods. Each metaheuristic was configured with the parameters listed in Table 1, which were selected because they allow the optimal performance to be reached according to their respective literature. This strategic configuration ensures that each combined approach utilizes optimized settings to effectively evaluate the detection capabilities of the MSER method enhanced by each metaheuristic algorithm. In Table 1, the FDBAEO and FDBSOS algorithms include a large number of parameters. Their values have been set to the same values suggested by their authors, as recorded in their own references.

Table 1. Parameter configuration of the metaheuristic techniques used in combination with MSER.

Algorithm	Parameters
PSO [47]	$N = 20, c_1 = 2, c_2 = 2, \omega = 0.4$
FF [48]	$N = 20, \gamma = 0.5, \beta_0 = 1, \alpha = 0.5$
GWO [49]	$N = 20, a = (2, 0)$
GA [50]	$N = 20, p_c = 0.9, p_m = 0.9$, selection method = roulette
FDBAEO [36]	The parameter values have been configured according to [r1]
FDBSOS [37]	The parameter values have been configured according to [r1]

In our comparative analysis, all metaheuristic methods were standardized to use the same number of search agents, specifically $N = 20$, to ensure fairness and consistency across different algorithm performances. All experiments for this study were carried out utilizing MATLAB R2022b as the main computational platform. These experiments were executed on a computer system that featured an Intel Core i7 processor operating at 2.60 GHz, alongside 32 GB of RAM and an RTX 2070 graphics processing unit.

The experimental results of this study are divided into two main sections to evaluate the detection methods comprehensively. In the first section (Section 5.1), we performed a visual comparison of the image results to assess their visual consistency and coherence. This qualitative analysis is essential for evaluating the perceptual quality and efficacy of each detection method to produce visually meaningful and consistent results. The main objective of this evaluation was to accurately detect anomalies associated with COVID-19 in the images. In the subsequent section (Section 5.2), we shift the focus to quantitative assessment, where we compare the numerical results derived from the experiments. Here, we employ well-established indices to assess the quality and accuracy of the detection outcomes. This quantitative approach provides objective performance metrics, enabling precise evaluation of the efficacy of each method based on key image quality indicators. By integrating both visual and numerical assessments, our goal was to provide a comprehensive evaluation of the detection methodologies being tested, presenting a balanced view of their overall capabilities.

5.1. Visual Evaluation

In this subsection, we conduct a visual analysis of the results obtained from each algorithm under review. Our primary goal in this comparative study is to explore the effectiveness of the detection methods in accurately identifying regions showing anomalies related to COVID-19. The core of this visual comparison is to evaluate whether these methods can coherently and meaningfully pinpoint the lung areas impacted by COVID-19. To aid in this examination, Figures 5 and 6 have been compiled to display the outcomes produced by the four algorithms across the five images used in our experiments. This visual representation allows for a direct and detailed assessment of the detection results, providing insights into each method's capability to accurately capture and delineate the affected regions within the images.

Figure 5 displays the results achieved by the MSER-PSO, MSER-FF, MSER-GWO, MSER-GA, MSER-FDBAEO, and MSER-FDBSOS detection methods over healthy images. The original images depicted do not exhibit completely opaque areas but rather showcase many details that obscure the visualization of the underlying lung structures. This characteristic is typically indicative of healthy lung tissue, which is a common feature observed in the absence of abnormalities linked to viral infections like COVID-19. In radiographs of healthy lungs, such as those illustrated in Figure 5, the areas appear less dense and opaque compared to what is seen in cases of ground-glass opacities, where lung tissue becomes more solid due to the accumulation of fluids and inflammatory cells. Given these conditions, none of the methods—MSER-PSO, MSER-FF, MSER-GWO, MSER-GA, MSER-FDBAEO, and MSER-FDBSOS—detect abnormalities indicative of lung infection.

Instead, all algorithms identify very small and granular regions that do not correspond to infected areas, demonstrating their current limitations in distinguishing affected lung tissue under these specific imaging conditions. In the images, the different colors represent the detected regions. Since in our method, the MSER algorithm, is executed many times until the optimal detection parameters are found, the regions appear with a different color, but the color has no special meaning.

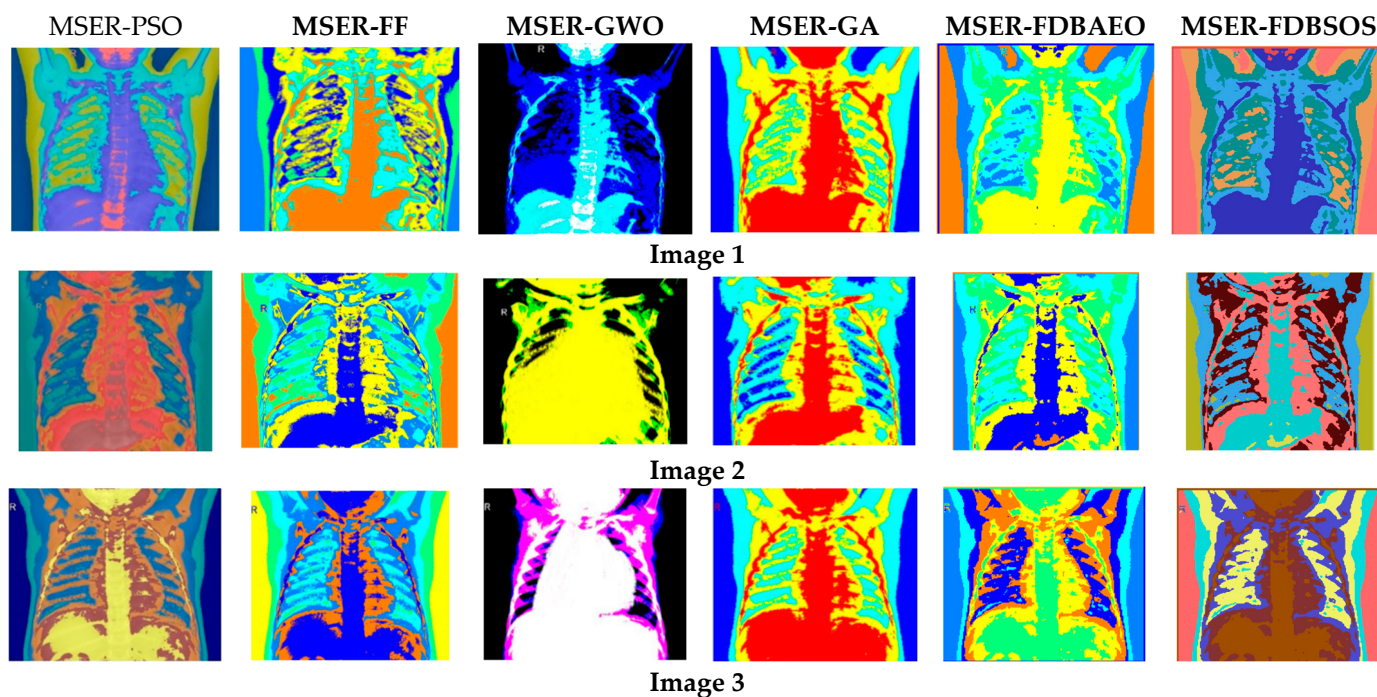


Figure 5. Results achieved by the MSER-PSO, MSER-FF, MSER-GWO, and MSER-GA detection methods over healthy images.

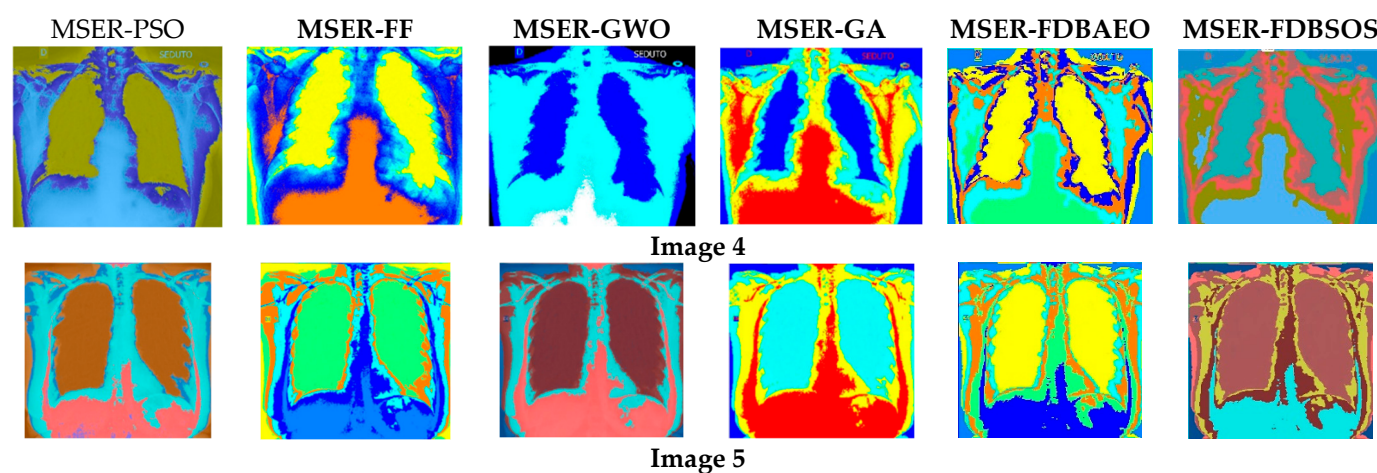


Figure 6. Results from the detection methods applied to radiographs of patients infected with COVID-19.

Figure 6 presents the outcomes from the detection methods applied to radiographs of patients infected with COVID-19. These images feature opaque areas, which allow for a complete visualization of the underlying lung structures—a characteristic commonly observed in viral infections such as COVID-19. These opacities signify regions of inflammation and fluid accumulation, where the lung tissue becomes denser and more solid due

to these pathological changes. Consequently, these areas exhibit increased homogeneity, particularly within the lung regions, due to the uniformity in density and opacity. Notably, all detection methods—MSER-PSO, MSER-FF, MSER-GWO, MSER-GA, MSER-FDBAEO, and MSER-FDBSOS—effectively identify the affected lung areas, confirming their utility in recognizing signs typical of COVID-19 infection. Among these, the MSER-FDBAEO and MSER-FDBSOS methods demonstrate the best visual results, showcasing their superior capability in accurately depicting the infected lung tissue in the provided radiographs.

5.2. Numerical Evaluation

Metaheuristic optimization algorithms are methods devised to solve optimization problems by deploying some specific search strategy, which could be very different from one algorithm to another. The differences in the search strategies deployed by each individual metaheuristic have always led to speculation about their performance over a wide range of different problems. There are many cases in the current literature where it has been demonstrated that some metaheuristics perform better over certain kinds of problems, whereas for others, they led to unfavorable results. When discussing the solutions delivered by metaheuristic optimization algorithms, one should always consider these as approximations to the optimal solution rather than a true optimal solution, and how well these solutions approximate the optimal solution is strongly dependent on how well a metaheuristic's search strategy adapts to a given problem. With that being said, the difference in the parameter configurations found by each algorithm lies in the “quality” of the resulting segmented image, which is evaluated in terms of the objective function presented in Section 4.2 (The Entropy/Cluster Quality Function “J”). As per our experiments, the FDBSOS metaheuristic is the one that shows better minimization results over the proposed problem; thus, it should be assumed that the parameter configuration found by this method is the best among those delivered by all tested algorithms. The experiments were run 40 times for each method, and these statistical results are shown in Table 2.

Table 2. Numerical results for the objective function.

		Image 1	Image 2	Image 3	Image 4	Image 5
FF	AVG	2.78×10^{-3}	2.74×10^{-3}	2.79×10^{-3}	2.73×10^{-3}	2.72×10^{-3}
	STD	2.76×10^{-3}	2.74×10^{-3}	2.79×10^{-3}	2.73×10^{-3}	2.72×10^{-3}
	MEDIAN	1.20×10^{-3}	1.10×10^{-3}	1.40×10^{-3}	1.00×10^{-3}	1.90×10^{-3}
GA	AVG	2.28×10^{-3}	2.17×10^{-3}	2.28×10^{-3}	2.28×10^{-3}	2.17×10^{-3}
	STD	1.76×10^{-4}	8.82×10^{-4}	1.76×10^{-4}	1.76×10^{-4}	8.82×10^{-4}
	MEDIAN	2.20×10^{-3}	2.10×10^{-3}	2.20×10^{-3}	2.20×10^{-3}	2.10×10^{-3}
PSO	AVG	2.78×10^{-3}	2.74×10^{-3}	2.79×10^{-3}	2.73×10^{-3}	2.72×10^{-3}
	STD	2.76×10^{-3}	2.74×10^{-3}	2.79×10^{-3}	2.73×10^{-3}	2.72×10^{-3}
	MEDIAN	1.20×10^{-3}	1.10×10^{-3}	1.40×10^{-3}	1.00×10^{-3}	1.90×10^{-3}
GWO	AVG	2.28×10^{-3}	2.17×10^{-3}	2.28×10^{-3}	2.28×10^{-3}	2.17×10^{-3}
	STD	1.76×10^{-4}	8.82×10^{-4}	1.76×10^{-4}	1.76×10^{-4}	8.82×10^{-4}
	MEDIAN	2.20×10^{-3}	2.10×10^{-3}	2.20×10^{-3}	2.20×10^{-3}	2.10×10^{-3}
FDBAEO	AVG	1.08×10^{-3}	1.08×10^{-3}	1.31×10^{-3}	1.31×10^{-3}	1.10×10^{-3}
	STD	1.10×10^{-4}	1.10×10^{-4}	2.79×10^{-5}	2.79×10^{-5}	1.10×10^{-4}
	MEDIAN	1.00×10^{-3}	1.00×10^{-3}	1.30×10^{-3}	1.30×10^{-3}	1.10×10^{-3}
FDBSOS	AVG	1.08×10^{-3}	1.31×10^{-3}	1.31×10^{-3}	1.31×10^{-3}	1.31×10^{-3}
	STD	1.10×10^{-4}	2.79×10^{-5}	2.79×10^{-5}	2.79×10^{-5}	2.79×10^{-5}
	MEDIAN	1.00×10^{-3}	1.30×10^{-3}	1.30×10^{-3}	1.30×10^{-3}	1.30×10^{-3}

To conduct a numerical assessment of the image detection results, we utilized a comprehensive suite of established quality indices, providing an objective measure of performance. This study incorporated a diverse array of metrics [52–55] to ensure a thorough evaluation of the detection quality and accuracy. The indices selected include the

Peak Signal-to-Noise Ratio (*PSNR*), Normalized Absolute Error (*NAE*), Normalized Cross Correlation (*NCC*), Structural Similarity Index (*SSIM*), Feature Similarity Index (*FSIM*), Universal Image Quality Index (*UIQI*), Quality Index Based on Local Variance (*QILV*), and Homogeneity Assessment of Region Roughness (*HARR*). Each of these metrics has been chosen for their demonstrated reliability in quantitatively and objectively assessing the quality of detection in images. By employing these metrics, we are able to precisely gauge the accuracy and fidelity of the detection outcomes, providing a solid and unbiased basis for evaluating the performance of the detection algorithms used in our study. This rigorous approach ensures that we can effectively compare and analyze the capabilities of different detection methods under consistent evaluation criteria.

To calculate the different indexes, an original image, defined as I_0 , is considered, whose dimensions are $M \times N$. This image is used as the input of all detection methods. As a result, an image I_D with the detected regions is generated. Various indexes are utilized to assess the discrepancies between the image that has been processed, I_D , with detected regions and an image referred to as the ground truth, I_{GT} , which is manually detected by human experts.

The Peak Signal-to-Noise Ratio (*PSNR*) is a widely accepted metric for evaluating the quality of detection in an image. It measures the amount of noise or distortion present in the image, in relation to ground truth image. A higher *PSNR* value indicates a lower degree of distortion or error, which suggests that the segmented image closely resembles the original. The *PSNR* is calculated as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^M \sum_{j=1}^N (I_{GT}(i, j) - I_D(i, j))^2}{M \cdot N}} \quad (12)$$

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

The performance index *NAE* is a quantitative metric used to assess the accuracy of image segmentation techniques. It evaluates the precision of detection by comparing the processed image I_D to a reference or ground truth image I_{GT} . This metric is particularly useful for determining how close the detection result is to the actual desired output. The value of the *NAE* is computed as follows:

$$NAE = \frac{\sum_{i=1}^M \sum_{j=1}^N (I_{GT}(i, j) - I_D(i, j))}{\sum_{i=1}^M \sum_{j=1}^N I_{GT}(i, j)} \quad (13)$$

The *NCC* is used to measure the similarity between two images—typically, the reference image (or ground truth I_{GT}) and the produced image I_D . The *NCC* is especially useful in scenarios where precise alignment and similarity to a reference image are crucial. The values of the *NCC* range from 0 to 1. Zero indicates that the detected regions in I_D do not align or correlate at all with the reference image I_{GT} .

The value of one indicates that I_D perfectly matches I_{GT} in terms of their spatial intensity patterns. The value of the *NCC* is determined as follows:

$$NCC = \frac{\sum_{i=1}^M \sum_{j=1}^N (I_{GT}(i, j) - \mu_{GT}) \cdot (I_D(i, j) - \mu_D)}{\sqrt{\sum_{i=1}^M \sum_{j=1}^N (I_{GT}(i, j) - \mu_{GT})^2 \cdot (I_D(i, j) - \mu_D)^2}} \quad (14)$$

where μ_{GT} and μ_D represent the mean values of the images I_{GT} and I_D , respectively. The *SSIM* method is a robust metric employed to evaluate the detection produced by the four methods in this study. The *SSIM* evaluates the structural similarity between the produced image I_D and a reference image I_{GT} . 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 processed and reference images, signifying superior segmentation quality. The *SSIM* was computed as follows:

$$SSIM = \frac{(2\mu_{GT}\mu_D + C1)(2\sigma_{GT}\sigma_D + C2)}{(\mu_{GT}^2 + \mu_D^2 + C1)(\sigma_{GT}^2 + \sigma_D^2 + C2)} \quad (15)$$

where σ_D and σ_{GT} corresponds to the standard deviation of images I_D and I_{GT} , respectively. $C1$ and $C2$ are two constants ($C1 = C2 = 0.065$). The *FSIM* served as a metric to assess the quality of the segmented images. The *FSIM* evaluates structural and textural information within the processed image, which is achieved by comparing it to a reference image. In contrast to conventional metrics that predominantly consider luminance and structural factors, the *FSIM* places a strong emphasis on capturing the perceptual quality of a segmented image. A higher *FSIM* score denotes a closer resemblance between the processed image and reference image, indicating superior segmentation quality concerning the preservation of vital visual features and textures.

The *UIQI* is a statistical metric used to evaluate the quality of a processed image by comparing its values with a reference or ground truth image. This index is also known as the Quality Index based on data correlation. The *UIQI* is calculated using the following formula:

$$UIQI = \frac{4\sigma_{D,GT}\mu_{GT}\mu_D}{(\mu_{GT}^2 + \mu_D^2)(\sigma_{GT}^2 + \sigma_D^2)} \quad (16)$$

where $\sigma_{D,GT}$ is the covariance between the processed I_D and ground truth I_{GT} images.

The *QILV* is designed to assess the quality of a processed image. This index is specifically useful for evaluating how well the detected regions preserve the local variance characteristics of the original image when compared to a reference or ground truth segmentation.

$$QILV = \frac{\sigma_{D,GT}}{\sqrt{\sigma_D^2 \sigma_{GT}^2}} \quad (17)$$

The performance index *HARR* is used to evaluate the homogeneity and roughness of the detected regions in an image. The *HARR* assesses how uniformly textures or patterns are represented within segmented areas and how smoothly or roughly these areas transition in the context of the entire image. A low *HAAR* value means better quality of detected images, while a high value means worse quality.

In the experiments, each metaheuristic algorithm is employed to calibrate the MSER method. Given that metaheuristic algorithms are based on stochastic principles, the solutions they find can vary slightly with each execution. This variability is a characteristic of metaheuristic approaches, highlighting their capacity to explore diverse areas of the search space and potentially discover different yet valid parameter configurations for optimal calibration of the MSER method. To reduce the stochastic effect, the results are analyzed after running each algorithm 30 times on the same image. Table 3 presents a comprehensive comparison of the numerical results obtained by applying four different algorithms to five images labeled 1–5. The numerical values correspond to the averages found during the 30 executions. The performance indices used to evaluate the algorithms include the *PSNR*, *NCC*, *SSIM*, *FSIM*, *UIQI*, and *QILV*, where a closer approximation to one indicates superior performance. Conversely, indices such as the *NAE* and *HARR* are more favorable when they approach zero, reflecting better accuracy in the detection process. According to the data summarized in Table 2, effective results across these indices suggest the detection of COVID-19. This inference is drawn from the ability of the MSER method to identify the lungs and other critical regions accurately, demonstrating a high degree of similarity between the algorithmically processed image and the reference image segmented by human experts. When the performance indices achieve better values, it indicates a probable diagnosis of COVID-19 in the patient. If the values are low, conversely, it suggests that the image is from a healthy patient without detectable anomalies.

Table 3. Numerical results obtained by applying four different algorithms to five images labeled 1–5.

	Image	PSNR	NAE	NCC	SSIM	FSIM	UIQI	QILV	HARR
MSER-PSO	Image 1	22.40	0.3972	0.8998	0.8320	0.8711	0.7884	0.6090	0.8420
	Image 2	20.20	0.3940	0.8888	0.8173	0.8412	0.7379	0.6830	0.8810
	Image 3	21.98	0.3820	0.7092	0.8244	0.8310	0.7988	0.6840	0.8822
	Image 4	24.16	0.2630	0.9863	0.9644	0.9827	0.9845	0.8988	0.4261
	Image 5	24.30	0.2872	0.9940	0.9442	0.9422	0.9884	0.8920	0.4620
	Averaged	22.60	0.3446	0.8956	0.8764	0.8936	0.8596	0.7533	0.6986
MSER-FF	Image 1	23.18	0.3794	0.8042	0.7920	0.8340	0.6699	0.6742	0.8800
	Image 2	23.99	0.4087	0.7820	0.7830	0.8720	0.7720	0.7640	0.8840
	Image 3	23.61	0.3864	0.7037	0.8940	0.8740	0.6637	0.7630	0.8840
	Image 4	23.63	0.2400	0.9040	0.8619	0.9199	0.8840	0.8740	0.5820
	Image 5	24.50	0.2409	0.8987	0.8849	0.9448	0.8687	0.8698	0.5920
	Averaged	22.68	0.3310	0.8185	0.8431	0.8888	0.7716	0.7890	0.7644
MSER-GWO	Image 1	20.46	0.7761	0.7888	0.7220	0.7453	0.8402	0.7998	0.7840
	Image 2	21.99	0.7412	0.7787	0.7440	0.7427	0.7300	0.7993	0.7920
	Image 3	21.79	0.7841	0.6740	0.7840	0.7445	0.8000	0.7994	0.7979
	Image 4	22.40	0.4027	0.8763	0.8903	0.8930	0.9020	0.8798	0.4244
	Image 5	23.79	0.4171	0.8277	0.8906	0.9442	0.9008	0.8977	0.4261
	Averaged	22.08	0.6242	0.7891	0.8061	0.8139	0.8346	0.8352	0.6448
MSER-GA	Image 1	19.40	0.5710	0.6420	0.6040	0.6390	0.6088	0.6990	0.8840
	Image 2	19.73	0.5900	0.5490	0.6900	0.6440	0.6370	0.6993	0.9920
	Image 3	19.21	0.5820	0.6240	0.5720	0.6420	0.6984	0.6883	0.8840
	Image 4	22.85	0.3720	0.7120	0.7900	0.8459	0.8084	0.8897	0.5720
	Image 5	22.37	0.3000	0.7962	0.8040	0.8666	0.8340	0.8999	0.5261
	Averaged	20.71	0.4830	0.6646	0.6920	0.7275	0.7173	0.7752	0.7716
MSER-FDBAEO	Image 1	23.20	0.4240	0.9087	0.9849	0.9428	0.7697	0.8198	0.7810
	Image 2	23.63	0.4200	0.9740	0.9619	0.9129	0.7840	0.8140	0.7220
	Image 3	23.61	0.4286	0.9937	0.9240	0.8720	0.8622	0.71830	0.7910
	Image 4	23.99	0.4108	0.9010	0.9430	0.9720	0.8300	0.7640	0.2820
	Image 5	23.18	0.4779	0.9042	0.9320	0.8320	0.8402	0.7742	0.4200
	Averaged	23.52	0.4322	0.9363	0.9491	0.9163	0.8172	0.7780	0.5992
MSER-FDBSOS	Image 1	23.10	0.3240	0.4787	0.8849	0.9428	0.8397	0.7998	0.8710
	Image 2	23.62	0.4200	0.8440	0.8719	0.9219	0.7242	0.7930	0.8420
	Image 3	23.61	0.4286	0.9737	0.9242	0.9740	0.8622	0.7842	0.8310
	Image 4	23.99	0.4108	0.8810	0.9830	0.9703	0.8600	0.8340	0.8420
	Image 5	23.18	0.3779	0.8042	0.8902	0.8220	0.8902	0.8942	0.8200
	Averaged	23.51	0.3922	0.7963	0.9108	0.9062	0.8352	0.8010	0.8412

The data presented in Table 3 clearly demonstrate that the MSER-FDBAEO and MSER-FDBSOS algorithms consistently outperform the other methods across a variety of evaluation metrics. Specifically, in terms of the *PSNR*, *NAE*, and *NCC* indices, these methods achieve the highest scores, indicating minimal distortion or error and thus a more accurate representation of the original image in the segmented version. Furthermore, the MSER-PSO and MSER-FF methods excel in the *SSIM* and *FSIM* indices, which evaluate the preservation of essential structural information that closely aligns with the reference image. This underscores their superior segmentation quality. In addition, MSER-FDBAEO and MSER-FDBSOS lead in the indices *UIQI*, *QILV*, and *HARR*, demonstrating an enhanced ability to maintain critical visual features and textures between the segmented and reference images, highlighting their advanced segmentation capabilities. These strong performance indicators collectively underscore the efficacy and reliability of the MSER-FDBAEO and MSER-FDBSOS methods in handling complex image segmentation tasks. In comparison, the MSER-PSO and MSER-FF methods rank second-best, showing commendable segmentation quality but not reaching the levels of MSER-FDBAEO and MSER-FDBSOS. Meanwhile, MSER-GA and MSER-GWO consistently perform the poorest, revealing significant short-

comings compared to the other methods. This comprehensive analysis not only highlights the superior performance of MSER-FDBAEO and MSER-FDBSOS but also provides valuable insights into the strengths and weaknesses of each method within the context of image detection, clearly illustrating the distinct advantages of each approach.

Three of the most important indices for measuring the quality of detected regions are *PSNR*, *SSIM*, and *FSIM*. Together, these metrics allow us to evaluate the image quality in terms of its main characteristics. To extend our analysis, we assessed the distribution of these three indices. Figure 7 illustrates these distributions for the (a) *PSNR*, (b) *SSIM*, and (c) *FSIM* indices. The distributions represent how each index is spread, with each algorithm's index calculated as an average over 30 independent runs on each image. According to Figure 7a, the MSER-FDBAEO and MSER-FDBSOS methods clearly exhibit the highest *PSNR* values, indicating superior performance compared to the other methods. Figure 7b shows that the MSER-FDBAEO and MSER-FDBSOS algorithms also achieve better distribution and higher values in terms of the *SSIM* index, outperforming the other methods. Similarly, Figure 7c demonstrates that the MSER-FDBAEO and MSER-FDBSOS algorithms maintain the best numerical values for the *FSIM* index. These results collectively indicate that the MSER-FDBAEO and MSER-FDBSOS methods consistently produce the best performance across the *PSNR*, *SSIM*, and *FSIM* indices. The MSER-PSO and MSER-FF methods rank second, while the MSER-GWO and MSER-GA methods show the poorest distribution and numerical values.

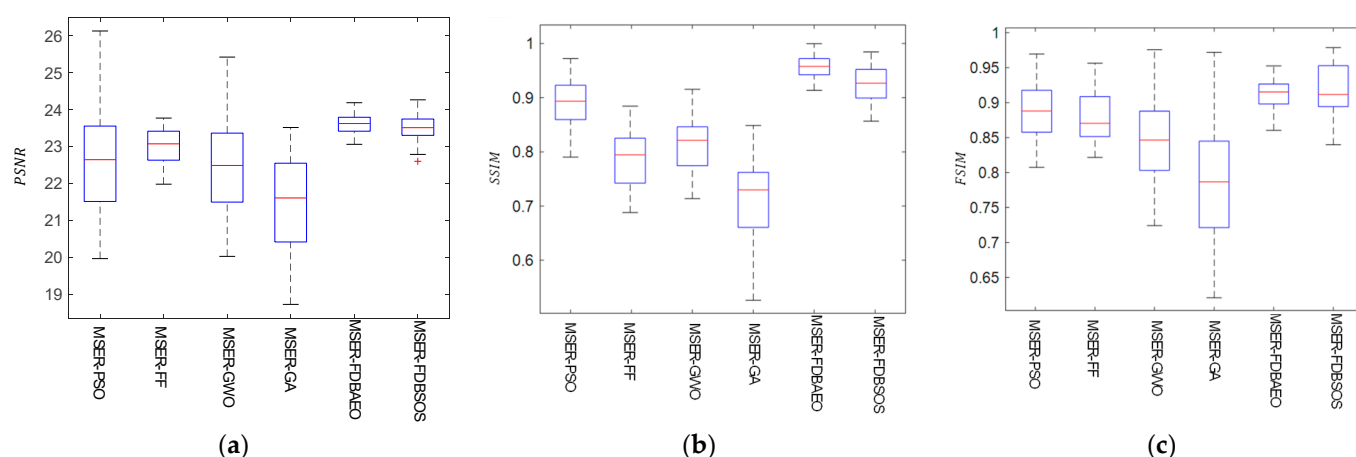


Figure 7. Distributions for the (a) *PSNR*, (b) *SSIM*, and (c) *FSIM* indices.

The outstanding performance of the MSER-FDBAEO and MSER-FDBSOS methods can be attributed to the robust detection capabilities inherent in the maximally stable extremal regions (MSER) technique. These capabilities are significantly enhanced by precisely tuning the MSER parameters through metaheuristic optimization methods. It is important to note that not all metaheuristic algorithms deliver the same level of performance, as their effectiveness largely depends on the specific nature of the optimization problem and the characteristics of the search space involved. Consequently, our experiments have demonstrated that the FDBAEO and FDBSOS algorithms are particularly well-suited for this application. These methods excel in adapting to the complexities of the search space associated with optimizing the MSER parameters, leading to more accurate and reliable identification of anomalies in the images. This suitability makes FDBAEO and FDBSOS preferred choices for enhancing the detection strength of the MSER method in identifying features associated with COVID-19 in X-ray images.

Among all the indices used in this study, the *FSIM* index stands out as one of the most important because it evaluates both the structural and textural information of the processed image. This means it effectively measures the overall quality of the image produced with the detected regions. To statistically evaluate the performance of the methodologies

compared in our study, we employed the Friedman ranking test [55]. The Friedman ranking test is a non-parametric statistical test, ideal for comparing and ranking multiple methods across various datasets or scenarios, especially when the normality assumption of the data is not met. This test provides a robust alternative to parametric tests like ANOVA. In the Friedman test, each method is ranked based on its performance within each dataset, with the best-performing method receiving a rank of 1, the second-best a rank of 2, and so on. If two methods exhibit identical performance, they are assigned the average of the ranks they would have occupied. Table 4 presents the results of the Friedman analysis, which takes into account the *FSIM* index values from Table 3. According to these results, the MSER-FDBAEO method achieves the best performance, followed by MSER-FDBSOS, while the other methods receive lower rankings, indicating poorer performance.

Table 4. Friedman test ranking results of the *FSIM* values from Table 3.

Algorithm	Ranking	Mean Value
MSER-FDBAEO	1	5.607121
MSER-FDBSOS	2	5.201473
MSER-PSO	3	4.892410
MSER-FF	4	4.552140
MSER-GWO	5	3.788974
MSER-GA	6	3.571401

6. Time Analysis and Accuracy

This section extends the analysis to evaluate the performance of integrating MSER with various metaheuristic algorithms. The evaluation includes a detailed discussion of the computational time required by MSER and each metaheuristic algorithm. Additionally, the accuracy of each approach is assessed in terms of its success rate. This comprehensive analysis will provide insights into both the efficiency and effectiveness of the combined methods, highlighting the trade-offs between computational cost and detection accuracy.

All metaheuristic techniques produce new solutions using a diverse range of processes, from straightforward to intricate, each with varying computational requirements. Due to their stochastic components and complex structures, traditional complexity analysis is not practical for metaheuristic algorithms. To evaluate the efficiency of each method, an experiment was conducted in which each method was run with MSER on each image. The time taken in seconds to obtain the result was recorded. To reduce the impact of randomness, each function's execution was repeated 30 times. The averaged inverted time values over all functions and their 30 executions are depicted in Figure 8. This analysis offers a comparative understanding of the computational efficiency of each method, emphasizing the differences in performance across different algorithms.

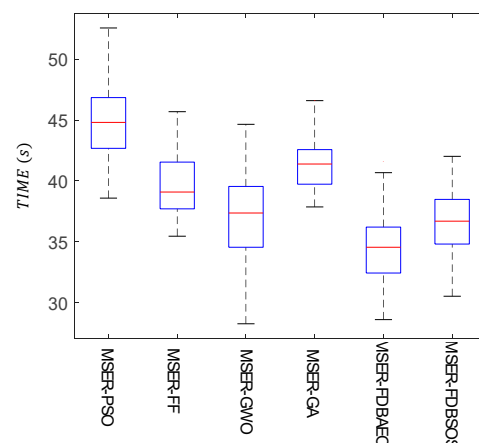


Figure 8. Averaged computational time invested by each approach.

An analysis of Figure 8 reveals that the MSER-FDBAEO and MSER-FDBSOS methods achieve the shortest processing times, ranging between 34 and 36 s. This indicates that these approaches have highly efficient search mechanisms, enabling swift exploration of the search space. In contrast, the MSER-FF, MSER-GWO, and MSER-GA methods exhibit average processing times between 38 and 42 s, indicating a moderate level of efficiency. Finally, the MSER-PSO method has the longest processing time, exceeding 45 s, suggesting that while it may offer high accuracy, it does so at the cost of greater computational effort.

The success rate (SR) is a critical metric used to evaluate the performance of a classification process. It is defined as the percentage of correctly classified instances out of the total instances evaluated. This metric provides a straightforward measure of the classifier's accuracy, indicating how well the model can identify the correct category or class for each instance. In the context of evaluating classification processes, the success rate plays a pivotal role as it directly reflects the model's effectiveness in making accurate predictions. A high success rate implies that the classifier reliably distinguishes between different classes, which is essential for applications where precise classification is crucial. Conversely, a low success rate suggests that the model struggles to correctly classify instances, indicating a need for further refinement or alternative approaches. Therefore, the success rate serves as a fundamental indicator of a classification system's overall performance and reliability.

Table 5 presents the success rate (SR) in the percentages of each metaheuristic approach combined with the MSER algorithm. According to the data, the MSER-FDBAEO and MSER-FDBSOS methods achieve the highest detection success rates for anomalous regions indicative of COVID-19. The MSER-PSO and MSER-FF methods exhibit average classification performance, while the MSER-GWO and MSER-GA methods have the lowest success rates. This indicates that MSER-FDBAEO and MSER-FDBSOS are the most effective in accurately identifying anomalous regions, whereas MSER-GWO and MSER-GA are less reliable in this classification task.

Table 5. The success rate (SR) in the percentages of each metaheuristic approach combined with the MSER algorithm.

Algorithm	SR
MSER-FDBAEO	94.25%
MSER-FDBSOS	92.14%
MSER-PSO	89.62%
MSER-FF	85.68%
MSER-GWO	70.11%
MSER-GA	65.74%

7. Conclusions

This study introduces a novel algorithm designed to detect abnormalities in X-ray images indicative of COVID-19 infection. This approach employs the maximally stable extremal regions (MSER) method for the precise detection of abnormal regions characteristic of the disease. To improve the accuracy of the MSER method, we employ an optimization process facilitated by a metaheuristic algorithm to select the optimal set of parameters for MSER. This optimization is achieved by examining various parameter configurations, where the effectiveness of each configuration is assessed based on an objective function that evaluates the entropy generated and the homogeneity within the detected regions.

We have evaluated our methodology utilizing a variety of metaheuristic algorithms—Particle Swarm Optimization (PSO), the Grey Wolf Optimizer (GWO), Firefly (FF), Genetic Algorithms (GA), artificial ecosystem optimization by means of fitness–distance balance (FDBAEO), and the standard symbiotic organism search with the fitness–distance balance method (FDBSOS)—to identify the most effective approach for our purposes. One of the significant advantages of our methodology is its efficiency; it does not necessitate the intensive process of collecting a large dataset of training images and meticulously labeling them to establish ideal outcomes. This is a notable characteristic of methods reliant on

deep learning, which typically require extensive datasets and substantial computational resources for training. Our approach, instead, considers the inherent capabilities of metaheuristic algorithms to optimize the parameters of our detection system directly, thereby simplifying the preparatory stages and accelerating the overall process of anomaly detection in images. This makes our methodology not only effective but also more accessible and less resource-intensive, offering a practical alternative in scenarios where rapid deployment and ease of implementation are crucial.

Our proposed method has been extensively tested across multiple images, with its detection quality evaluated using several indices. The results from these tests show that our algorithm has a strong ability to support healthcare professionals in accurately detecting COVID-19, demonstrating its great potential and effectiveness.

Future research should explore a broader range of metaheuristic methods to determine which ones most effectively calibrate the MSER algorithm for detecting anomalous regions. Expanding the variety of metaheuristic algorithms tested could uncover more efficient and accurate calibration techniques. Additionally, extending the study to include more refined analytical tools would enhance the evaluation of each metaheuristic method's capabilities. These advanced analytical elements would provide deeper insights into the effectiveness of different methods in improving region detection, ultimately leading to more precise and reliable outcomes.

Author Contributions: Conceptualization, A.G.; methodology, F.F. and M.P.-C.; software, V.G.-G.; validation, A.G., E.C. and F.F.; formal analysis, A.G., E.C. and F.F.; investigation, V.G.-G. and M.P.-C.; resources, V.G.-G. and M.P.-C.; data curation, V.G.-G. and M.P.-C.; writing—original draft preparation, V.G.-G., A.G., E.C. and F.F.; writing—review and editing, A.G., E.C. and F.F.; visualization, A.G., E.C. and F.F.; supervision, A.G.; project administration, A.G.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data available on request from the authors.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Mishra, N.P.; Das, S.S.; Yadav, S.; Khan, W.; Afzal, M.; Alarifi, A.; Kenawy, E.-R.; Ansari, M.T.; Hasnain, S.; Nayak, A.K. Global impacts of pre- and post-COVID-19 pandemic: Focus on socio-economic consequences. *Sens. Int.* **2020**, *1*, 100042. [\[CrossRef\]](#) [\[PubMed\]](#)
- van Mulukom, V.; Pummerer, L.J.; Alper, S.; Bai, H.; Čavojová, V.; Farias, J.; Kay, C.S.; Lazarevic, L.B.; Lobato, E.J.; Marinthe, G.; et al. Antecedents and consequences of COVID-19 conspiracy beliefs: A systematic review. *Soc. Sci. Med.* **2022**, *301*, 114912. [\[CrossRef\]](#) [\[PubMed\]](#)
- Pascarella, G.; Strumia, A.; Piliego, C.; Bruno, F.; Del Buono, R.; Costa, F.; Scarlata, S.; Agrò, F.E. COVID-19 diagnosis and management: A comprehensive review. *J. Intern. Med.* **2020**, *288*, 192–206. [\[CrossRef\]](#) [\[PubMed\]](#)
- Zhai, P.; Ding, Y.; Wu, X.; Long, J.; Zhong, Y.; Li, Y. The epidemiology, diagnosis and treatment of COVID-19. *Int. J. Antimicrob. Agents* **2020**, *55*, 105955. [\[CrossRef\]](#) [\[PubMed\]](#)
- Jamshaid, H.; Zahid, F.; Din, I.U.; Zeb, A.; Choi, H.G.; Khan, G.M.; Din, F.U. Diagnostic and Treatment Strategies for COVID-19. *AAPS Pharmscitech* **2020**, *21*, 222. [\[CrossRef\]](#) [\[PubMed\]](#)
- Rahman, T.; Khandakar, A.; Qiblawey, Y.; Tahir, A.; Kiranyaz, S.; Kashem, S.B.A.; Islam, M.T.; Al Maadeed, S.; Zughaier, S.M.; Khan, M.S.; et al. Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images. *Comput. Biol. Med.* **2021**, *132*, 104319. [\[CrossRef\]](#) [\[PubMed\]](#)
- Feki, I.; Ammar, S.; Kessentini, Y.; Muhammad, K. Federated learning for COVID-19 screening from Chest X-ray images. *Appl. Soft Comput.* **2021**, *106*, 107330. [\[CrossRef\]](#) [\[PubMed\]](#)
- López-Cabrera, J.D.; Orozco-Morales, R.; Portal-Díaz, J.A.; Lovelle-Enríquez, O.; Pérez-Díaz, M. Current limitations to identify COVID-19 using artificial intelligence with chest X-ray imaging. *Health Technol.* **2021**, *11*, 411–424. [\[CrossRef\]](#) [\[PubMed\]](#)
- Ahishali, M.; Degerli, A.; Yamac, M.; Kiranyaz, S.; Chowdhury, M.E.H.; Hameed, K.; Hamid, T.; Mazhar, R.; Gabbouj, M. Advance Warning Methodologies for COVID-19 Using Chest X-Ray Images. *IEEE Access* **2021**, *9*, 41052–41065. [\[CrossRef\]](#)
- Benmalek, E.; Elmhamdi, J.; Jilbab, A. Comparing CT scan and chest X-ray imaging for COVID-19 diagnosis. *Biomed. Eng. Adv.* **2021**, *1*, 100003. [\[CrossRef\]](#)
- Shen, J.; Zhang, C.J.P.; Jiang, B.; Chen, J.; Song, J.; Liu, Z.; He, Z.; Wong, S.Y.; Fang, P.-H.; Ming, W.-K. Artificial Intelligence Versus Clinicians in Disease Diagnosis: Systematic Review. *JMIR Public Heal. Surveill.* **2019**, *7*, e10010. [\[CrossRef\]](#) [\[PubMed\]](#)

12. Enshaeei, N.; Oikonomou, A.; Rafiee, M.J.; Afshar, P.; Heidarian, S.; Mohammadi, A.; Plataniotis, K.N.; Naderkhani, F. COVID-rate: An automated framework for segmentation of COVID-19 lesions from chest CT images. *Sci. Rep.* **2022**, *12*, 3212. [[CrossRef](#)] [[PubMed](#)]
13. Arun Prakash, J.; Asswin, C.R.; Dharshan Kumar, K.S.; Dora, A.; Ravi, V.; Sowmya, V.; Gopalakrishnan, E.A.; Soman, K.P. Transfer learning approach for pediatric pneumonia diagnosis using channel attention deep CNN architectures. *Eng. Appl. Artif. Intell.* **2023**, *123*, 106416. [[CrossRef](#)]
14. Kaya, Y.; Gürsoy, E. A MobileNet-based CNN model with a novel fine-tuning mechanism for COVID-19 infection detection. *Soft Comput.* **2023**, *27*, 5521–5535.
15. Chen, S.; Ren, S.; Wang, G.; Huang, M.; Xue, C. Interpretable CNN-Multilevel Attention Transformer for Rapid Recognition of Pneumonia From Chest X-Ray Images. *IEEE J. Biomed. Health Informatics* **2023**, *28*, 753–764. [[CrossRef](#)] [[PubMed](#)]
16. Sanghvi, H.A.; Patel, R.H.; Agarwal, A.; Gupta, S.; Sawhney, V.; Pandya, A.S. A deep learning approach for classification of COVID and pneumonia using DenseNet-201. *Int. J. Imaging Syst. Technol.* **2022**, *33*, 18–38. [[CrossRef](#)] [[PubMed](#)]
17. AnbuDevi, M.K.A.; Suganthi, K. Review of Semantic Segmentation of Medical Images Using Modified Architectures of UNET. *Diagnostics* **2022**, *12*, 3064. [[CrossRef](#)] [[PubMed](#)]
18. Zhou, Z.; Siddiquee, M.M.R.; Tajbakhsh, N.; Liang, J. UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation. *IEEE Trans. Med. Imaging* **2019**, *39*, 1856–1867. [[CrossRef](#)] [[PubMed](#)]
19. Diakogiannis, F.I.; Waldner, F.; Caccetta, P.; Wu, C. ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data. *ISPRS J. Photogramm. Remote Sens.* **2020**, *162*, 94–114. [[CrossRef](#)]
20. Li, X.; Chen, H.; Qi, X.; Dou, Q.; Fu, C.W.; Heng, P.A. H-DenseUNet: Hybrid densely connected UNet for liver and tumor segmentation from CT volumes. *IEEE Trans. Med. Imaging* **2018**, *37*, 2663–2674. [[CrossRef](#)]
21. Aiadi, O.; Khaldi, B. A fast lightweight network for the discrimination of COVID-19 and pulmonary diseases. *Biomed. Signal Process. Control.* **2022**, *78*, 103925. [[CrossRef](#)]
22. Basu, A.; Sheikh, K.H.; Cuevas, E.; Sarkar, R. COVID-19 detection from CT scans using a two-stage framework. *Expert Syst. Appl.* **2021**, *193*, 116377. [[CrossRef](#)] [[PubMed](#)]
23. Xue, X.; Chinnaperumal, S.; Abdulsahib, G.M.; Manyam, R.R.; Marappan, R.; Raju, S.K.; Khalaf, O.I. Design and Analysis of a Deep Learning Ensemble Framework Model for the Detection of COVID-19 and Pneumonia Using Large-Scale CT Scan and X-ray Image Datasets. *Bioengineering* **2023**, *10*, 363. [[CrossRef](#)] [[PubMed](#)]
24. Cuevas, E.; Rodríguez, A. *Metaheuristic Computation with MATLAB®*; Informa UK Limited: London, UK, 2020.
25. Ztürk, H.T.; Kahraman, H.T. Meta-heuristic search algorithms in truss optimization: Research on stability and complexity analyses. *Appl. Soft Comput.* **2023**, *145*, 110573. [[CrossRef](#)]
26. Riaz, M.; Bashir, M.; Younas, I. Metaheuristics based COVID-19 detection using medical images: A review. *Comput. Biol. Med.* **2022**, *144*, 105344. [[CrossRef](#)] [[PubMed](#)]
27. Júnior, D.A.D.; da Cruz, L.B.; Diniz, J.O.B.; da Silva, G.L.F.; Junior, G.B.; Silva, A.C.; de Paiva, A.C.; Nunes, R.A.; Gattass, M. Automatic method for classifying COVID-19 patients based on chest X-ray images, using deep features and PSO-optimized XGBoost. *Expert Syst. Appl.* **2021**, *183*, 115452. [[CrossRef](#)] [[PubMed](#)]
28. Gopatoti, A.; Vijayalakshmi, P. CXGNet: A tri-phase chest X-ray image classification for COVID-19 diagnosis using deep CNN with enhanced grey-wolf optimizer. *Biomed. Signal Process. Control* **2022**, *77*, 103860. [[CrossRef](#)] [[PubMed](#)]
29. Khadija, B. Automatic detection of COVID-19 using CNN model combined with Firefly algorithm. In Proceedings of the 2022 8th International Conference on Optimization and Applications (ICOA), Sestri Levante, Italy, 6–7 October 2022; pp. 1–4.
30. Gülmez, B. A novel deep neural network model based Xception and genetic algorithm for detection of COVID-19 from X-ray images. *Ann. Oper. Res.* **2023**, *328*, 617–641. [[CrossRef](#)]
31. Naik, M.K.; Swain, M.; Panda, R.; Abraham, A. Novel Square Error Minimization-Based Multilevel Thresholding Method for COVID-19 X-Ray Image Analysis Using Fast Cuckoo Search. *Int. J. Image Graph.* **2024**, *24*, 2450004. [[CrossRef](#)]
32. Ramkumar, M.; Gowtham, M.; Jamaesha, S.S.; Vigenesh, M. Attention induced multi-head convolutional neural network organization with MobileNetv1 transfer learning and COVID-19 diagnosis using jellyfish search optimization process on chest X-ray images. *Biomed. Signal Process. Control.* **2024**, *93*, 106133. [[CrossRef](#)]
33. Liu, B.; Nie, X.; Li, Z.; Yang, S.; Tian, Y. Evolving deep convolutional neural networks by IP-based marine predator algorithm for COVID-19 diagnosis using chest CT scans. *J. Ambient. Intell. Humaniz. Comput.* **2024**, *15*, 451–464. [[CrossRef](#)] [[PubMed](#)]
34. Rachel, R.B.; Nehemiah, H.K.; Singh, V.K.; Manoharan, R.M.V. Diagnosis of Covid-19 from CT slices using Whale Optimization Algorithm, Support Vector Machine and Multi-Layer Perceptron. *J. X-ray Sci. Technol.* **2024**, *32*, 253–269. [[CrossRef](#)]
35. Mahdy, A.; Shaheen, A.; El-Sehiemy, R.; Ginidi, A. Artificial ecosystem optimization by means of fitness distance balance model for engineering design optimization. *J. Supercomput.* **2023**, *79*, 18021–18052. [[CrossRef](#)]
36. Kahraman, H.T.; Aras, S.; Gedikli, E. Fitness-distance balance (FDB): A new selection method for meta-heuristic search algorithms. *Knowledge-Based Syst.* **2020**, *190*, 105169. [[CrossRef](#)]
37. Nistér, D.; Stewénus, H. Linear time maximally stable extremal regions. In Proceedings of the Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, 12–18 October 2008; Proceedings, Part II 10. Springer: Berlin/Heidelberg, Germany, 2008; pp. 183–196.
38. Lee, M.H.; Park, I.K. Performance evaluation of local descriptors for maximally stable extremal regions. *J. Vis. Commun. Image Represent.* **2017**, *47*, 62–72. [[CrossRef](#)]

39. Matas, J.; Chum, O.; Urban, M.; Pajdla, T. Robust wide-baseline stereo from maximally stable extremal regions. *Image Vis. Comput.* **2004**, *22*, 761–767. [[CrossRef](#)]
40. Martins, P.; Carvalho, P.; Gatta, C. On the completeness of feature-driven maximally stable extremal regions. *Pattern Recognit. Lett.* **2016**, *74*, 9–16. [[CrossRef](#)]
41. Chang, H.; Li, D.; Zhang, X.; Cui, X.; Fu, Z.; Chen, X.; Song, Y. Real-time height measurement with a line-structured-light based imaging system. *Sensors Actuators A Phys.* **2024**, *368*, 115164. [[CrossRef](#)]
42. Arul, U.; Arun, V.; Rao, T.P.; Baskaran, R.; Kirubakaran, S.; Hussan, M.I.T. Effective Anomaly Identification in Surveillance Videos Based on Adaptive Recurrent Neural Network. *J. Electr. Eng. Technol.* **2024**, *19*, 1793–1805. [[CrossRef](#)]
43. Atlas, L.G.; Arjun, K.; Kumar, K.S.; Dhanaraj, R.K.; Nayyar, A. Deep learning enabled hemorrhage detection in retina with DPFE and splat segmentation in fundus images. *Biomed. Signal Process. Control.* **2024**, *88*, 105692. [[CrossRef](#)]
44. Tohidi, N.; Rustamov, R.B. Short overview of advanced metaheuristic methods. *Int. J. Tech. Phys. Probl. Eng. (IJTPE)* **2022**, *14*, 84–97.
45. Joyce, T.; Herrmann, J.M. A review of no free lunch theorems, and their implications for metaheuristic optimisation. *Nat. -Inspired Algorithms Appl. Optim.* **2018**, *744*, 27–51.
46. Eberhart, R.C.; Kennedy, J. A new optimizer using particle swarm theory. In Proceedings of the 6th International Symposium on Micro Machine and Human Science 1995, Nagoya, Japan, 13–16 March 1995; pp. 39–43.
47. Johari, N.F.; Zain, A.M.; Noorfa, M.H.; Udin, A. Firefly Algorithm for Optimization Problem. *Appl. Mech. Mater.* **2013**, *421*, 512–517. [[CrossRef](#)]
48. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [[CrossRef](#)]
49. Lambora, A.; Gupta, K.; Chopra, K. Genetic algorithm-A literature review. In Proceedings of the 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 14–16 February 2019; pp. 380–384.
50. Zhang, H.; Fritts, J.E.; Goldman, S.A. Entropy-based objective evaluation method for image segmentation. In *Storage and Retrieval Methods and Applications for Multimedia 2004*; SPIE: Bellingham, WA, USA, 2003; Volume 5307, pp. 38–49.
51. Ben-David, S.; Ackerman, M. Measures of clustering quality: A working set of axioms for clustering. In Proceedings of the Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–11 December 2008; pp. 121–128.
52. Müller, D.; Soto-Rey, I.; Kramer, F. Towards a guideline for evaluation metrics in medical image segmentation. *BMC Res. Notes* **2022**, *15*, 210. [[CrossRef](#)]
53. Taha, A.A.; Hanbury, A.; del Toro, O.A.J. A formal method for selecting evaluation metrics for image segmentation. In Proceedings of the 2014 IEEE International Conference on Image Processing (ICIP), Paris, France, 27–30 October 2014; pp. 932–936.
54. Kumar, S.N.; Lenin Fred, A.; Ajay Kumar, H.; Sebastin Varghese, P. Performance metric evaluation of segmentation algorithms for gold standard medical images. In *Recent Findings in Intelligent Computing Techniques: Proceedings of the 5th ICACNI 2017, Goa, India, 1–3 June 2017*; Springer: Singapore, 2018; Volume 3, pp. 457–469.
55. Friedman, M. A Comparison of Alternative Tests of Significance for the Problem of m Rankings. *Ann. Math. Stat.* **1940**, *11*, 86–92. Available online: <http://www.jstor.org/stable/2235971> (accessed on 1 June 2024). [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.