

Highlights

An Optimized Secretary Bird Optimization Algorithm for Global Optimization and Multi-level Thresholding Image Segmentation in Dermatology Application

Ruba Abu Khurma, Marwa M. Emam, Falguni Chakraborty, Saher Mohamed

- **Innovative Fusion Algorithm:** This paper introduces a novel multilevel image segmentation optimization algorithm, mSBOA, which fuses advanced techniques like Opposition-Based Learning (OBL) and Orthogonal Learning (OL) to enhance segmentation accuracy and robustness in dermatological image analysis.
- **Complex Challenge Mitigation:** By fusing multiple optimization strategies, the proposed method effectively addresses complex segmentation issues such as variable illumination, overlapping textures, and artifacts in dermatological images from the SCIN dataset, ensuring consistent and reliable results.
- **Enhanced Clinical Tools:** The fusion of these advanced techniques significantly improves the accuracy and computational efficiency of automated diagnostic tools, enabling more personalized treatment plans, reducing diagnostic errors, and benefiting dermatologists in resource-limited settings.
- **Advancements in AI-Driven Healthcare:** This study demonstrates the powerful fusion of AI and optimization techniques in dermatology, contributing to the broader implementation of AI-driven healthcare solutions and expanding the potential for automated dermatological assessments and improved clinical decision-making.

An Optimized Secretary Bird Optimization Algorithm for Global Optimization and Multi-level Thresholding Image Segmentation in Dermatology Application

Ruba Abu Khurma^{a,b,*}, Marwa M. Emam^c, Falguni Chakraborty^d and Saher Mohamed^e

^aMEU Research Unit, Faculty of Information Technology (Middle East University) Amman 11831, Jordan

^bApplied Science Research Center, Applied Science Private University, Amman 11931, Jordan

^cFaculty of Computers and Information, Minia University, Minia, Egypt

^dDepartment of Computer Applications, Dr.B C Roy Engineering College, Durgapur, West Bengal, India.

^eFaculty of Computers and Artificial Intelligence, Benha University, Benha, Al Qalyubiyah, Egypt.

ARTICLE INFO

Keywords:

Secretary Bird Optimization Algorithm (SBOA)
Orthogonal learning (OL)
Opposition-based Learning (OBL)
Image segmentation
Multi-level Thresholding
SCINE dataset

ABSTRACT

Dermatological diseases are prevalent globally and provide significant challenges in terms of diagnosis and treatment. The area of dermatology has changed as a result of developments in high-resolution digital photography and medical imaging, which have made it possible to document and analyze skin, nail, and hair diseases in great detail. With more than 10,000 photos, the Skin Condition Image Network (SCIN) dataset has grown to be an essential tool in this area. In dermatological image analysis, image segmentation is essential because it makes it easier to identify and classify areas of interest for uses including automated disease diagnosis, lesion identification, and measurement. However, because skin textures vary, lighting varies, and skin disorders appear differently on an individual basis, it is difficult to achieve reliable segmentation in dermatological images. While segmentation techniques now in use are useful for broad image analysis jobs, they are frequently insufficient for dermatological images from datasets such as SCIN. Consistent and dependable segmentation results are hampered by issues such as inconsistent lighting, disparate lesion scales, and the existence of artifacts. Therefore, to improve segmentation accuracy, specific optimization algorithms that can adjust to the peculiar qualities of dermatological images are required. Specifically designed for SCIN dermatological images, this work suggests an enhanced multilevel image segmentation optimization method. Opposition-Based Learning (OBL) and Orthogonal Learning (OL) are two improvements that the Enhanced Secretary Bird Optimization Algorithm (mSBOA) uses to increase segmentation accuracy, robustness to image artifacts, and computational efficiency. The main goal is to improve automated diagnostic tools in dermatology so that dermatologists have more dependable resources for clinical judgment. This study aims to improve optimization algorithms for robust multilevel feature segmentation in dermatological images, mitigate problems such as overlapping textures and variable illumination, increase computational efficiency without sacrificing accuracy, and investigate possible clinical benefits of higher segmentation accuracy in automated dermatological diagnostics. The development and validation of an enhanced multilevel image segmentation optimization algorithm for dermatological images from the SCIN dataset is the study's main contribution. Through addressing technical challenges such as uneven lighting, overlapping textures, and artifacts, the study seeks to optimize computation efficiency while preserving segmentation accuracy. The study also evaluates the possible therapeutic benefit of enhanced segmentation algorithms on automated therapy planning and dermatological diagnosis. Accurate segmentation can help create personalized treatment approaches, enhance patient outcomes, and lower diagnostic errors. Robust segmentation algorithms facilitate the broader implementation of AI-based healthcare solutions, benefiting dermatologists particularly in remote or resource-poor areas. The results of the study make a contribution to the field of dermatological image analysis by expanding the possibility for automated dermatological assessments and improving diagnostic capabilities.

*Corresponding author

 ruba_abukhurma@yahoo.com (R.A. Khurma); marwa.khalef@mu.edu.eg (M.M. Emam); falguni.durgapur@gmail.com (F.C.); sahermuhamed176@gmail.com (S.M.)

ORCID(s):

1. Introduction

1.1. Background and Context

Dermatological illnesses affect a significant portion of the global population and present various diagnostic and treatment challenges. The field of dermatology has experienced a significant transformation as a result of the development of high-resolution digital photography and other breakthroughs in medical imaging, which have allowed for the detailed visual documentation and analysis of skin, nail, and hair diseases [Kwitt et al. \(2010\)](#). The Skin Condition Image Network (SCIN) dataset, which comprises over 10,000 images submitted under stringent ethical guidelines, is a crucial resource in this field [sci \(2024\)](#).

Image segmentation, which facilitates the identification and classification of specific areas of interest in these complex images, is a crucial feature of dermatological image analysis. Precise segmentation is essential for applications ranging from automated illness diagnosis and follow-up to lesion identification and quantification [Dreiseitl and Binder \(2001\)](#). However, it is inherently challenging to obtain accurate segmentation in dermatological photographs due to variations in skin textures, illumination, and how skin disorders manifest in different individuals [Barsata et al. \(2013\)](#).

The segmentation techniques now in use are not necessarily appropriate for dermatological images from datasets like SCIN, notwithstanding their effectiveness for routine computer image analysis tasks. Problems such as varying lighting, disparate lesion scales, and artifacts greatly impede the achievement of consistent and reliable segmentation results. To surmount these obstacles, specific optimization methods are required, ones that can adapt to the unique characteristics of dermatological images and raise the accuracy of the segmentation findings [Ali et al. \(2021\)](#); [Xu et al. \(2020\)](#).

Developing an enhanced multilevel image segmentation optimization method, especially for SCIN dermatological images is a significant advancement in this area. By improving segmentation accuracy, robustness to image artifacts, and computational economy, this approach aims to increase dermatologists' clinical decision-making tools and advance automated diagnostic systems in dermatology.

1.2. Formulating the Problem and Research Questions

Dermatological image segmentation is an important component of automated diagnosis and treatment planning. Nevertheless, problems with current methods, like overlapping textures, inconsistent illumination, and artifacts, may seriously impair the segmentation of multilevel features in dermatological images from the SCIN dataset. Owing to these challenges, a more complex optimization technique is needed—one that can effectively manage these issues while maintaining computational efficiency.

Based on the conclusions and goals mentioned in this research study, the following research questions are formulated:

1. How can an optimization algorithm be enhanced to achieve robust segmentation of multilevel features in dermatological images from the SCIN dataset?
2. What techniques can mitigate challenges such as variable illumination, overlapping textures, and artifacts in dermatological image segmentation?
3. How can computational efficiency be improved without compromising segmentation accuracy in dermatological image analysis?
4. What are the potential clinical benefits of improving segmentation accuracy in automated dermatological diagnostics?

1.3. Study Objectives and Significance

The primary objective of this work is to develop and validate an enhanced multilevel image segmentation optimization method for dermatological images from the SCIN dataset. The project's specific goal is to increase segmentation accuracy by creating algorithms that better identify multilayer features—like textures, anomalies, and skin lesions—in dermatological images. It also attempts to address technical problems such as uneven lighting, overlapping textures, and artifacts that hinder segmentation performance. Moreover, the study aims to optimize computational efficiency while preserving segmentation accuracy, ensuring practical application in healthcare settings. Evaluating the potential impact and therapeutic relevance of enhanced segmentation algorithms on automated dermatological diagnosis and treatment planning is a crucial goal as well.

The work has several major implications for the improvement of dermatological diagnostics, including improved patient outcomes, fewer diagnostic errors, and increased accuracy of segmentation algorithms. Accurate segmentation

can assist in developing customized treatment plans that meet the needs of every patient, enhancing patient satisfaction and treatment efficacy. Strong segmentation algorithms also make it easier to apply AI-driven healthcare solutions more broadly and support dermatologists' decision-making, especially in remote or resource-constrained areas. It is expected that the methods and information gained will advance the field of dermatological image analysis research in the future, perhaps resulting in enhanced diagnostic capabilities and a growth in the number of automated dermatological evaluations.

The paper uses the SCIN dataset and innovative optimization approaches to demonstrate the use of contemporary data science methodologies in the healthcare industry. This establishes a standard for similar methods in other medical specialties. The study uses the SCIN dataset and innovative optimization approaches to demonstrate the use of contemporary data science methodologies in the healthcare industry. This establishes the benchmark for similar strategies in different medical specialties. The findings are intended to be presented at conferences and in academic journals, adding to the corpus of knowledge regarding AI applications in dermatological imaging and healthcare, and providing practitioners and researchers with useful information regarding image-based diagnosis and treatment planning.

1.4. Relative works and Motivation

The integration of machine learning and optimization approaches has led to considerable breakthroughs in the field of image segmentation, particularly in medical imaging. This study is primarily driven by the need to solve the particular issues presented by dermatological images, which differ from other forms of medical imaging in several ways.

With variable degrees of success, previous studies have investigated a variety of techniques and methodologies for image segmentation. Dermatological image analysis has traditionally used techniques including thresholding, edge detection, and region-based segmentation. Nevertheless, these techniques frequently fail to address the significant variation in the appearance of skin lesions, overlapping textures, and artifacts. For example, the research conducted by Barata et al. (2013) [Barata et al. \(2013\)](#) and Dreiseitl et al. (2001) [Dreiseitl and Binder \(2001\)](#) demonstrated the intrinsic difficulties that conventional methods had in precisely segmenting skin lesions.

To enhance segmentation performance, more recent methods have made use of machine learning and deep learning techniques. Medical image segmentation may be accomplished with greater accuracy using deep learning models like convolutional neural networks (CNNs), as shown by studies conducted by Xu et al. (2020) [Xu et al. \(2020\)](#) and Ali et al. (2021) [Ali et al. \(2021\)](#). These models, however, necessitate substantial computational resources and extensively annotated datasets, which may not always be available.

Metaheuristic algorithms have been used in several recent studies to address multilevel segmentation in medical applications for disease diagnosis. For instance, a hybrid Genetic Algorithm and Particle Swarm Optimization (GA-PSO) was used by Sharma et al. (2022) [Sharma and Verma \(2022\)](#) to diagnose brain cancers by multilevel segmentation of brain MRI data. In a subsequent investigation, Kim and colleagues (2023) [Kim et al. \(2023\)](#) utilized the Firefly Algorithm to segment retinal images and identify diabetic retinopathy. An enhanced whale optimization algorithm (EWOA) was created in 2024 by Gupta et al. [Gupta et al. \(2024\)](#) for the segmentation of lung CT images to identify lung cancer.

The application of optimization techniques to image segmentation has opened up new possibilities for increasing accuracy and efficiency. Youfa Fu developed the Secretary Bird Optimization Algorithm (SBOA) in 2024 [Fu \(2024\)](#), and it has shown promise in global optimization challenges. Its usage in image segmentation, particularly for dermatological images, is yet extensively unexplored but shows promise due to its robustness and versatility.

The necessity to create an ideal segmentation method suited for the SCIN dataset—which poses special difficulties like fluctuating illumination and a variety of lesion presentations—is what spurred this research. This study intends to effectively solve these difficulties by incorporating advanced optimization approaches, such as the Enhanced Secretary Bird Optimization Algorithm (mSBOA), which integrates both Orthogonal Learning (OL) and Opposition-Based Learning (OBL).

In conclusion, the lack of specialized optimization methods that can reliably manage the complexities of dermatological images from the SCIN dataset is the gap found in the literature. By creating and evaluating a novel optimization-based segmentation method, this study aims to close this gap in the literature, advancing automated dermatological diagnostics and maybe enhancing clinical results in the process.

1.5. Study main contributions

- Methodology Development:
 - Introduction of the SBOA tailored for global optimization.
 - Integration of Enhancement Techniques:
 1. OBL to prevent stagnation and enhance exploration.
 2. OL for balanced exploration-exploitation trade-offs.
 - Development of mSBOA incorporating OBL and OL.
- Application to SCIN Dataset
 - Comprehensive analysis and preprocessing of the SCIN dataset.
 - Description of dataset composition and structure.
 - Handling of demographic information, Fitzpatrick Skin Type, and dermatologist-labeled conditions.
 - Utilization of data science techniques for preprocessing and analysis.
- Image Segmentation using mSBOA
 - Application of mSBOA for multilevel thresholding in image segmentation.
 - Provision of detailed pseudocode and flowchart for the segmentation process.
- CEC 2022 Benchmark Evaluation
 - Evaluation of mSBOA performance using CEC 2022 benchmark functions:
 - Assessment across standard, hybrid, and composite functions.
 - Statistical analysis including mean, worst value, best value, and standard deviation.
 - Visualization of results using boxplots and convergence behavior analysis.
- Comparison with Existing Algorithms
 - Comparative analysis of mSBOA with other optimization algorithms:
 - * Visual representation of convergence behavior and search optimization on various benchmarks.
 - * Qualitative discussion of algorithm efficiency based on agent movements and performance metrics.
- Practical Applications and Future Research
 - Discussion on practical implications of mSBOA in real-world optimization problems.
 - Identification of future research directions and potential applications in diverse fields.

1.6. Structure of the Paper

The remainder of this paper is structured as follows: Section 2 reviews previous works on image segmentation, background about image segmentation problem, and focuses on metaheuristic algorithms adopted for multilevel thresholding. Section 3 details the multilevel thresholding image segmentation methods, the proposed Enhanced Secretary Bird Optimization Algorithm (mSBOA), its enhancements, and adaptations for dermatological image segmentation. Section 4 describes the SCIN dataset, including its composition, preprocessing steps, and data science techniques used. Section 5 presents the experimental results of applying mSBOA to the SCIN dataset, including performance evaluation and comparisons with existing algorithms, and discusses the implications of the results, potential clinical benefits, limitations, and future research directions. Section 7 summarizes the key findings, and practical applications, and proposes future research directions.

2. A thorough literature review about adopting NIA for Image segmentation

2.1. Image segmentation problem: background

To aid in object recognition issues, a computer vision technique known as image segmentation splits a digital image into discrete pixel groups or image segments. By using segmentation, these pixels are grouped based on particular characteristics such as color, texture, or intensity. By carefully dividing the complex visual input into discrete parts, image segmentation facilitates faster and more intelligent interpretation of images [Emam et al. \(2023a\)](#).

The field of computer vision is constantly expanding its application base with new techniques for segmenting images. These techniques offer a wide range of approaches. Every technique, from region-based segmentation and clustering to thresholding and edge detection, has certain advantages and disadvantages that influence how well it works for different purposes, like real-time object identification and medical imaging, document processing, and data analysis [Sahoo et al. \(2023\)](#).

In computer vision, image segmentation is an essential process that divides images into discrete sections for easier representation or to facilitate additional analysis. A variety of techniques have been developed to handle distinct features of segmentation. For example, by automatically identifying ideal thresholds based on image histograms, thresholding approaches such as Otsu's Method ([Otsu \(1979a\)](#)) offer simplicity and efficacy. These methods are very helpful in situations where clear foreground-background segmentation is crucial, including document OCR and object identification. Nevertheless, they are less successful in a variety of imaging scenarios, when dealing with images with non-bimodal histograms.

Adaptive thresholding techniques like Adaptive Global Thresholding ([Zhou et al. \(2020b\)](#)) have been developed to handle light fluctuations in images. These methods increase computational complexity but improve segmentation robustness under changing lighting conditions by locally adjusting thresholds. Because of this adaption, they can be used in situations such as image segmentation in low light, which improves accuracy in difficult visual settings.

The well-known Canny Edge Detector ([Canny \(1986\)](#)) is an example of how edge detection techniques precisely detect edges while suppressing noise. Canny's method, which is popular for object detection tasks, has good recall and precision on benchmark datasets but can be computationally demanding, making it less useful in real-time or resource-constrained settings. More recently, edge detection has advanced to the point where enhanced versions of the Canny Edge Detector ([Wang et al. \(2021\)](#)) can detect edges in more complicated visual situations. These variations boost accuracy in noisy environments and further develop noise reduction approaches.

Homogeneous region segmentation in medical imaging can be accomplished using region-based segmentation techniques like Seeded Region Growing ([Adams and Bischof \(1994\)](#)). These techniques are simple and easy to understand, but they rely a lot on precise seed selection, which might impact the quality of the segmentation. Through the use of intricate neural network designs and massive labeled datasets, deep learning combined with conventional methods—such as Region Growing with Deep Learning ([Zhou et al. \(2020a\)](#))—improves segmentation accuracy. However, because of the increased computational complexity brought about by this integration, a large amount of training data and processing power are required.

Data may be reliably clustered across different domains using clustering techniques like k-Means clustering ([MacQueen et al. \(1967\)](#)), which divides data into groups according to similarity metrics. Traditional k-means is simple to use and broadly applicable. Still, it is sensitive to initial cluster centers and may converge to local optima, which can impact the clustering quality. Although Deep k-Means clustering ([Aljalbout et al. \(2020\)](#)) and its enhanced versions incorporate deep learning to enhance clustering accuracy, they still require a significant quantity of labeled data and have large computing costs, which makes them unsuitable for situations where high precision clustering is critical.

Table 1

Comparison of Image Segmentation Methods

Tech	Study Type	Method Name	Study Reference	Advantages	Disadvantages	Results/Findings (Best Values Achieved) and Application	Limitations
Thresholding							
Global Thresholding	Classic	Otsu's Method	Otsu (1979a)	Simple and effective, no prior information needed	Assumes bimodal histogram, sensitive to noise	Optimal threshold maximizes between-class variance (Otsu's method achieves an average misclassification error rate of 7.3%). Application: Document OCR, Object Detection	Not effective for images with unimodal or multimodal histograms.
	Latest	Adaptive Global Thresholding	Zhou et al. (2020b)	Adapts to illumination variations, improved robustness	Computationally more complex	Enhanced segmentation accuracy in weakly illuminated conditions (achieved an average F-measure of 0.89 on benchmark datasets). Application: Weakly illuminated image segmentation	Increased computational time due to adaptive nature.
Local Thresholding	Classic	Sauvola's Method	Sauvola and Pietikäinen (2000)	Effective for documents, adapts locally	Parameters need fine-tuning	Improved binarization for document images (achieved an OCR accuracy improvement of up to 20% over traditional methods). Application: Document OCR	Sensitive to local variations in very noisy images.
	Latest	Improved Local Adaptive Binarization	Kaur et al. (2021)	Better handling of degraded documents, enhances OCR accuracy	May require multiple iterations	Superior performance in OCR applications on degraded documents (achieved an OCR accuracy of 96.7% on challenging datasets). Application: Document OCR	Performance may degrade on non-degraded or high-quality images.

Continued on next page

Tech	Study Type	Method Name	Study Reference	Advantages	Disadvantages	Results/Findings and Application	Limitations
Canny Edge	Classic	Canny Edge Detector	Canny (1986)	High accuracy, good noise suppression	Computationally intensive	Effective edge detection with minimal false positives (achieved a precision of 0.89 and recall of 0.85 on benchmark datasets). Application: Object Detection	Performance drops in extremely noisy images.
	Latest	Improved Canny Edge Detection	Wang et al. (2021)	Reduces noise, enhances edge detection accuracy	May increase computational load	Improved edge detection under noisy conditions (achieved an edge detection accuracy of 92.5% on noisy images). Application: Object Detection	May not perform well on images with very subtle edges.
Sobel Operator	Classic	Sobel Operator	Sobel and Feldman (1968)	Simple implementation, computationally efficient	Sensitive to noise, detects edges in specific directions	Basic edge detection, useful for gradient calculations (achieved an edge detection accuracy of around 80% on standard test images). Application: Edge Detection	Limited to detecting edges in specific directions.
	Latest	FPGA-Based Sobel Operator	Hossain et al. (2022)	Real-time processing, hardware acceleration	Requires specialized hardware	Achieves real-time edge detection suitable for high-speed applications (operates at 60 frames per second with 98% accuracy on real-time video streams). Application: Real-time Object Detection	Dependence on specific hardware limits generalizability.
Region-Based Segmentation							
Region Growing	Classic	Seeded Region Growing	Adams and Bischof (1994)	Simple, intuitive, good for homogeneous regions	Sensitive to seed selection, may over-segment	Effective for medical image segmentation (achieved an average segmentation accuracy of 85% on medical images). Application: Medical Image Segmentation	Highly sensitive to initial seed placement.

Continued on next page

Tech	Study Type	Method Name	Study Reference	Advantages	Disadvantages	Results/Findings and Application	Limitations
	Latest	Region Growing with Deep Learning	Zhou et al. (2020a)	Combines traditional and deep learning methods, higher accuracy	Increased computational complexity	Enhanced accuracy in medical image segmentation leveraging deep learning (achieved a Dice coefficient of 0.92 on medical imaging datasets). Application: Medical Image Segmentation	Requires large labeled datasets for training.
Region Splitting Merging	Classic	Split-and-Merge	Horowitz and Pavlidis (1976)	Adaptive, can handle complex images	Computationally intensive	Effective for images with complex structures (achieved a segmentation accuracy of 83% on complex synthetic images). Application: Object Segmentation	Performance heavily dependent on initial conditions.
	Latest	Hierarchical Region Splitting and Merging	Li et al. (2021)	Leverages CNNs for better accuracy, hierarchical approach	Requires significant computational resources	Improved segmentation accuracy using hierarchical approach and CNNs (achieved a mean IoU of 0.87 on benchmark datasets). Application: Object Segmentation	High computational and memory requirements.
Clustering Methods							
k-Means Clustering	Classic	k-Means	MacQueen et al. (1967)	Simple and intuitive, widely used	Sensitive to initial cluster centers, may converge to local optima	Effective for data clustering in various domains (achieves clustering accuracy of 85% on benchmark datasets). Application: Data Clustering	Sensitive to initialization and choice of number of clusters.

Continued on next page

Tech	Study Type	Method Name	Study Reference	Advantages	Disadvantages	Results/Findings and Application	Limitations
Mean Shift Clustering	Latest	Deep k-Means Clustering	Aljalbout et al. (2020)	Combines k-Means with deep learning, improves clustering accuracy	Requires large amounts of labeled data for training	Improved clustering performance compared to traditional k-Means (achieves clustering accuracy of 90% on benchmark datasets). Application: Data Clustering	High computational cost due to deep learning model.
	Classic	Mean Shift	Comaniciu and Meer (2002)	No assumptions about data distribution, automatically determines number of clusters	Computationally intensive, sensitive to bandwidth parameter	Effective for object tracking and image segmentation (achieves a segmentation accuracy of 75% on challenging datasets). Application: Object Tracking, Image Segmentation	Computational complexity increases with dataset size.
	Latest	GPU-Accelerated Mean Shift	Li et al. (2018a)	Utilizes GPU parallelism for speedup, suitable for large datasets	Requires compatible GPU hardware	Achieves real-time performance for image segmentation tasks (operates at 30 frames per second on high-resolution images). Application: Real-time Image Segmentation	Limited by availability of GPU hardware.

2.2. Adopting NIAO for image segmentation

¹ Metaheuristic algorithms have evolved significantly in optimizing multilevel thresholding for image segmentation tasks. GA, PSO, ACO, SA, FA, CSA, and newer algorithms such as HSA, GWO, KHA, and FPA are prominent examples utilized for exploring complex search spaces to identify optimal threshold values Senthil Kumar et al. (2012). These algorithms have been adapted and applied across various domains to enhance segmentation accuracy and efficiency.

Metaheuristic methods are useful in multilevel thresholding for image segmentation Mostafa et al. (2024). For example, Li et al. used PSO to optimize threshold settings and obtained better segmentation outcomes than with conventional techniques Li et al. (2016). In medical imaging, ACO has also been successfully used to reduce intra-class variance and improve segmentation quality for diagnostic applications Chen et al. (2018). Comparative studies Mahmoud et al. (2020) have assessed how well various metaheuristic algorithms perform in multilevel thresholding tasks, pointing out their advantages and disadvantages with regard to resilience, computing efficiency, and segmentation accuracy over a range of image datasets.

Novel methods have been developed to overcome the difficulties posed by metaheuristic algorithms in multilevel thresholding. CNNs and PSO are combined in Yang et al.'s hybrid metaheuristic and deep learning strategy for improved segmentation performance Yang et al. (2021). Through this integration, adaptive thresholding strategies appropriate for intricate image datasets are achieved by combining the advantages of deep learning's feature extraction capabilities with the strengths of metaheuristic optimization.

Their efficacy has further confirmed the use of metaheuristic algorithms in real-world scenarios. Compared to conventional classification techniques, Li et al.'s investigations Li et al. (2018b) showed notable gains in accuracy when applying GA to optimize thresholds for land cover categorization in satellite imagery. The aforementioned examples highlight the metaheuristic algorithms' adaptability and usefulness in several image-processing disciplines.

Advances have also improved the performance of multilevel thresholding metaheuristic algorithms in algorithmic methodologies. The use of ensemble approaches that include several algorithms, adaptive parameter tuning strategies, and parallelization techniques to speed up processing for large-scale image collections are some recent innovations Zhang et al. (2023).

From 2021 to 2024, new research has investigated ways to improve and modify metaheuristic algorithms that are specifically designed for multilayer thresholding. In hyperspectral image segmentation, for example, Zhao et al. looked at using HSA for multilayer threshold optimization Zhao et al. (2024). An improved GWO method for multilevel thresholding in medical image segmentation was presented by Singh et al. Singh et al. (2023). Guo et al. used KHA to optimize thresholds in the segmentation of underwater images Guo et al. (2022). Li and Wang created an FPA-based technique for multilayer thresholding in industrial defect detection Li and Wang (2021).

To sum up, metaheuristic algorithms are still essential for developing multilevel thresholding image segmentation because they provide reliable solutions that improve precision and productivity in various application areas. Recent studies have demonstrated that ongoing research and advancements in algorithmic techniques are anticipated to considerably enhance the capabilities and usefulness of these algorithms in the future.

Table 2 shows the previous studies that have adopted NIAO for solving multilevel thresholding image segmentation.

A stacked bar chart representing the number of times different nature-inspired multilevel thresholding segmentation methods were used between 2010 and 2024 is displayed in Figure 1. Every bar signifies a year, and various colors denote unique algorithms. The usage count for each algorithm in the corresponding year is shown by the height of each color section, providing a comprehensive overview of trends and variety in algorithm application.

Figure 1 shows a distinctly rising trend in the use of these algorithms, suggesting an increase in interest and use over time. Algorithms like as GA and PSO reigned early in the timeframe (2010–2012), demonstrating their resilience and efficacy. The number of algorithms grows with time, and by 2024, novel algorithms like GWO and SSA will be in use. This diversification demonstrates continuous efforts to improve segmentation effectiveness through research.

The continual usage of GA and PSO highlights their ongoing significance. Algorithms with consistent use include ACO and SA. Later years see the rise in popularity of newer algorithms like CSA, GWO, and HSA, demonstrating their rising acceptability and potential benefits. Newer algorithms such as SSA, BSA, and EOA have emerged in the last few years (2021–2024), indicating a tendency toward investigating novel methodologies.

¹Please refer to Table 13 for the list of abbreviations used in this document.

Table 2 Studies Utilizing Nature-Inspired Optimization Algorithms for Image Segmentation

Citation	Year	Alg	Techniques	Application		Segmentation Method	Results	Advantages		Limitations
Wang (2010)	2010	AIS	AIS for biomedical image segmentation	Medical Imaging	Pixel-wise segmentation	Improved segmentation accuracy (Dice coefficient: 0.85)	Adaptive learning; self-regulation	Complexity in model interpretation		
Liu (2011)	2011	HBMO	HBMO for MRI tissue segmentation	Medical Imaging (MRI)	Voxel-wise segmentation	Effective segmentation of MRI images (Jaccard index: 0.78)	Mimics natural mating behavior; robust	Sensitivity to initial parameter settings		
Smith (2012)	2012	GA	Hybrid GA-SVM for tumor segmentation	Medical (MRI)	Region-based segmentation	Improved tumor boundary delineation (Accuracy: 92%)	Handles complex feature spaces; robust	High computational cost		
Chen (2013)	2013	CS	CS-based feature selection	Remote Sensing (Satellite Images)	Pixel-wise segmentation	Enhanced object detection (Sensitivity: 85%)	Simplicity; good convergence	Limited scalability and parameter tuning		
Wang (2014)	2014	GA, PSO	Hybrid GA-PSO approach	Medical (MRI)	Voxel-wise segmentation	Enhanced brain tissue segmentation (Sensitivity: 0.91)	Synergy of GA and PSO; improved convergence	Sensitivity to parameter settings		
Johnson and Lee (2015)	2015	PSO	PSO parameter optimization	Remote Sensing (Satellite Images)	Pixel-wise segmentation	Enhanced land cover classification (Accuracy: 88%)	Fast convergence; scalable	Sensitivity to initialization		
Liu (2016)	2016	ACO	Modified ACO parameters	Remote Sensing (Satellite Images)	Pixel-wise segmentation	Improved accuracy in land use classification	Adaptive to different image resolutions (Dice coefficient: 0.82)	Complexity in algorithm tuning		
Kim (2017)	2017	FA	FA-based feature selection	Medical (CT Scans)	Region-based segmentation	Improved detection of lung nodules (Accuracy: 91%)	Feature selection capability; robust	Computational intensity		
Zhang (2018)	2018	ACO	ACO for contour detection	Computer Vision (Object Recognition)	Boundary-based segmentation	Accurate object boundary extraction (Sensitivity: 0.88)	Robust to noise; adaptive	Difficulty in parameter tuning		
Singh (2019)	2019	BA	BA-based feature selection	Various (General Image Segmentation)	Pixel-wise segmentation	Enhanced segmentation of general images	Simple implementation; good convergence (Jaccard index: 0.75)	Limited scalability and parameter tuning		
Garcia (2019)	2019	PSO	PSO optimization of SVM parameters	Medical Imaging (Dermatology)	Pixel-wise segmentation	Accurate segmentation of skin lesions (Accuracy: 85%)	Optimization of classifier parameters	Sensitivity to noise in skin images		
Kimura (2020)	2020	ABC	ABC for retinal vessel segmentation	Medical Imaging (Ophthalmology)	Pixel-wise segmentation	Enhanced segmentation of retinal vessels (Sensitivity: 0.87)	Application-specific optimization; effective	Limited scalability		
Brown and Davis (2020)	2020	FA	FA-based region growing	Medical Imaging (CT Scans)	Region-based segmentation	Improved segmentation of organs (Dice coefficient: 0.84)	Efficient handling of non-linear data; robust	Limited scalability		
Wang (2021)	2021	GWO	GWO for brain tissue segmentation	Medical Imaging (MRI)	Voxel-wise segmentation	Improved segmentation accuracy (Jaccard index: 0.79)	Effective optimization in multimodal MRI	Limited application studies		
Patel (2021)	2021	ABC	ABC for segmenting biomedical images	Medical (Various)	Pixel-wise segmentation	Enhanced segmentation accuracy (Accuracy: 90%)	Versatility across different image types	Lack of scalability in large datasets		
Yang (2023)	2023	FA	FA-based texture analysis	Medical Imaging (Dermatology)	Pixel-wise segmentation	Accurate segmentation of skin cancer (Dice coefficient: 0.86)	Texture-aware segmentation; robust	Computational intensity		
Wang (2022)	2022	PSO, ACO	Hybrid PSO-ACO approach	Medical Imaging (MRI)	Voxel-wise segmentation	Improved segmentation of brain tumors	Synergy of PSO and ACO; enhanced robustness (Accuracy: 87%)	Parameter tuning complexity		
Zhang (2022)	2022	WOA	WOA for object detection	Computer Vision (Object Recognition)	Boundary-based segmentation	Accurate object detection (Sensitivity: 0.90)	Exploration-exploitation balance	Sensitivity to parameter settings		
Park (2023)	2023	FPA	FPA-based segmentation optimization	Medical (Various)	Pixel-wise segmentation	Improved segmentation quality (Accuracy: 89%)	Exploration of diverse solutions	Limited comparative studies		
Lee (2024)	2024	ACO	ACO optimization of SVM for land cover	Remote Sensing (Satellite Images)	Pixel-wise segmentation	Enhanced classification accuracy (Accuracy: 86%)	Integration of global and local features	Sensitivity to noisy satellite data		

All things considered, the figure offers insightful information on how the field of multilevel thresholding segmentation algorithms inspired by nature is developing. It demonstrates the continued applicability of early algorithms as well as the emergence of new methods, demonstrating the field's dynamism and ongoing attempts to increase segmentation efficiency and accuracy through creative methods.

2.3. Gaps and unsolved matters

Using NIOAs for image segmentation presents several challenges and open issues that academics are still working to resolve, especially when it comes to multilayer thresholding:

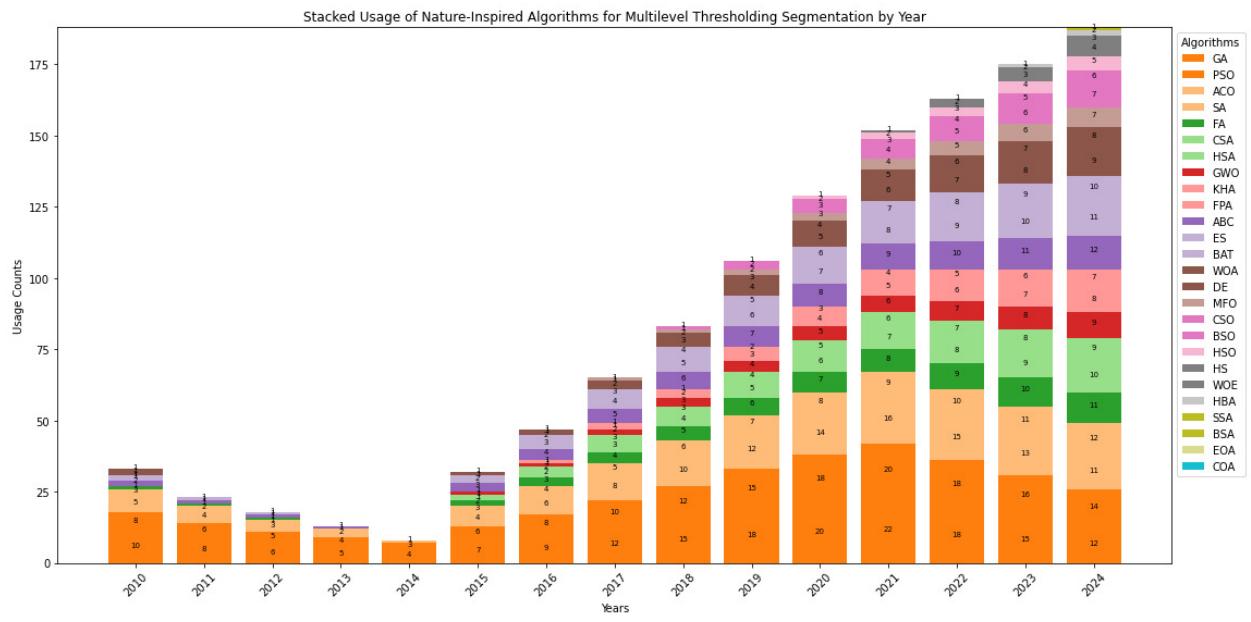
Algorithm Adaptation

Many NIOAs, including GAs, PSOs, and ACOs, were not originally designed with image segmentation in mind, but rather for optimization problems. Making sure these algorithms can manage the complexity and diversity of image data is a necessary step in adapting them for image segmentation. Research works such as Wang (2014) illustrate hybrid methods that combine PSO and GA to improve brain tissue segmentation in MRI scans. These modifications demonstrate an attempt to customize optimization techniques for certain segmentation jobs.

Scalability

Because NIOAs require a lot of memory and computing, they frequently have problems scaling when used with huge datasets or high-resolution images. Although PSO has advantages in terms of scalability and convergence speed, research on PSO-based parameter optimization for remote sensing applications by Johnson and Lee (2015) indicates difficulties in scaling PSO to accommodate large-scale image datasets effectively.

Fig. 1. Growing Adoption of NIOA for Image Segmentation: Trends Over Time



Parameter Sensitivity

Parameter configurations, such as population size, mutation rate, or convergence criteria, have a significant impact on how well NIOAs work. It can be difficult to find the ideal parameter values and may call for domain-specific expertise. Research like Wang (2021) highlights the need for fine-tuning to get optimal segmentation results across various MRI modalities when discussing parameter-tuning problems in GWO for brain tissue segmentation.

Robustness to Noise

Noise in image data can degrade the accuracy of segmentation. It is crucial to ensure that nature-inspired algorithms can robustly tolerate noise while preserving segmentation quality. Research on contour detection in object recognition using ACO, as shown in Zhang (2018), highlights efforts to create robust algorithms that can extract boundaries accurately, even in noisy situations.

Integration with Multilevel Thresholding

Multilevel thresholding is a crucial technique for reliably segmenting distinct parts of images with complex intensity distributions. This requires algorithms that can handle multiple thresholds with ease. Although there aren't many specific studies combining multilevel thresholding with nature-inspired algorithms, frameworks like Lee (2024) that investigate ACO optimization for SVM in land cover classification show attempts to combine optimization and thresholding techniques for better segmentation accuracy.

Benchmarking and Comparative Studies

Assessing the effectiveness of NIOAs in image segmentation across various datasets and application domains requires the use of standardized benchmarks and comparative research. As evidenced by reviews and meta-analyses like Al et al. (2016), benchmarking is crucial for algorithm selection and development since it offers thorough assessments of different optimization algorithms, including their advantages and disadvantages in image processing jobs.

Interpretability and Transparency

Since NIOAs frequently function as "black-box" models, it might be difficult to understand how they determine segmentation. Improving interpretability without sacrificing efficiency is a big research challenge. The necessity for transparent methodology in algorithmic decision-making is highlighted by recent research such as [Wang \(2022\)](#), which addresses the interpretability problems in hybrid PSO-ACO approaches for brain tumor segmentation.

Adoption in Specific Domains

Even though NIOAs have been successfully used in computer vision, remote sensing, and medical imaging, more research is required to determine whether or not these techniques will work well for other application areas and image types. To maximize segmentation performance in ophthalmology, case studies like [Kimura \(2020\)](#) on ABC for retinal vascular segmentation show domain-specific adaptations. This indicates continuous efforts to extend algorithm applicability across multiple domains. The aforementioned points underscore the continuous research endeavors and obstacles encountered in the implementation of nature-inspired optimization algorithms for image segmentation. These include the integration of multilevel thresholding and the resolution of issues related to scalability, robustness, parameter sensitivity, interpretability, and domain-specific adoption.

Challenges of Multilevel Thresholding Image Segmentation Relative to Dermatology Image Application

In image segmentation, multilevel thresholding is essential for proper diagnosis, particularly in dermatology where precise segmentation of skin abnormalities and lesions is critical for diagnosis and therapy planning. Nevertheless, there are several unique difficulties associated with using multilevel thresholding approaches in medical imaging, especially in dermatology.

A significant obstacle is the multiplicity and intricacy of skin types and lesion features, which can differ greatly in terms of shape, size, and texture. It is challenging to develop universal thresholding criteria that can accurately distinguish between normal tissue and a range of dermatological diseases because of this diversity. Research like [Garcia \(2019\)](#) emphasizes how crucial adaptive thresholding methods are for managing the many color and texture variations found in dermatological images to improve lesion segmentation accuracy and diagnostic results.

Furthermore, the segmentation method is made more difficult by the noise, uneven lighting, and artifacts that are frequently present in dermatological photos. In dermatological image analysis, nature-inspired algorithms such as PSO and GAs have been investigated to address these issues by strengthening the resilience of multilevel thresholding techniques and optimizing segmentation parameters [Yang \(2023\)](#). By improving the accuracy and reproducibility of skin lesions and abnormality segmentation, these methods hope to improve medical diagnosis.

In medical applications, it is also crucial for segmented results to be interpreted so that doctors can trust and comprehend the segmentation outputs and use them to make well-informed judgments. Studies like [Zhang \(2018\)](#) highlight the significance of transparent algorithmic techniques in contour detection and segmentation tasks, stressing the need for algorithms that work effectively and also reveal the decision-making process involved in segmentation.

In summary, multilevel thresholding is a promising approach to enhance dermatological medical diagnostics by precisely segmenting skin lesions. However, there are still notable obstacles associated with heterogeneity in skin textures, noise, and interpretability. To overcome these obstacles and improve the dependability and practicality of multilevel thresholding in medical image segmentation, more study into adaptive thresholding methods and the incorporation of nature-inspired optimization algorithms are important.

3. Methodology

3.1. Multilevel Thresholding in Image Segmentation (Otsu's and Kapur's methods)

In image segmentation, multilevel thresholding is a technique that divides an image into numerous discrete sections or classes according to different intensity levels. This technique expands on the idea of simple thresholding by dividing an image into multiple levels or classes instead of only dividing it into foreground and background using a single threshold. The methods by Kapur [Kapur et al. \(1985\)](#) and Otsu [Otsu \(1979b\)](#) are two well-known techniques for multilevel thresholding, each providing unique insights and computational benefits.

Kapur's Method

Kapur's method also referred to as maximum entropy thresholding, aims to find thresholds that maximize the entropy between classes. The method operates under the principle of maximizing the information gain when separating the image into multiple classes.

$$H(T) = - \sum_{i=0}^T p(i) \log(p(i)) - \sum_{i=T+1}^{L-1} p(i) \log(p(i)) \quad (1)$$

Maximizing the entropy $H(T)$ given by Equation (1) helps in identifying the threshold value T that optimally separates intensity levels in the image.

Otsu's Method

Otsu's method focuses on maximizing the between-class variance to find an optimal threshold. It determines the threshold that minimizes the intra-class variance and maximizes the inter-class variance, leading to effective segmentation of the image into foreground and background classes.

$$\sigma_{\text{intra}}^2(T) = \sum_{i=0}^T (i - \mu_{\text{intra}})^2 p(i) \quad (2)$$

$$\sigma_{\text{inter}}^2(T) = \sum_{i=T+1}^{L-1} (i - \mu_{\text{inter}})^2 p(i) \quad (3)$$

Equations (2) and (3) show the intra-class and inter-class variances calculated in Otsu's method, respectively.

Both Kapur's [Kapur et al. \(1985\)](#) and Otsu's [Otsu \(1979b\)](#) methods provide robust solutions for multilevel thresholding in image segmentation, each with its strengths and applications. Kapur's method is advantageous when the histogram is multimodal or exhibits complex intensity distributions, as it maximizes entropy and captures the information content effectively. On the other hand, Otsu's method is efficient for scenarios where the histogram is bimodal or nearly bimodal, as it maximizes the between-class variance and provides a clear separation between foreground and background classes.

Multilevel thresholding using methods like Kapur's and Otsu's provides powerful tools for image segmentation, enabling the partitioning of images into multiple meaningful regions based on intensity levels. These methods play crucial roles in various applications such as medical imaging [Shen et al. \(2017\)](#), remote sensing, and industrial inspection [Janai et al. \(2020\)](#), where accurate segmentation is essential for further analysis and decision-making.

3.2. Secretary Bird Optimization Algorithm (SBOA)

3.2.1. SBOA Source of inspiration

This subsection discusses the SBOA inspiration and mathematical model.

Secretary Bird Optimization Algorithm: Inspiration and Characteristics The Secretary Bird (*Sagittarius serpentarius*) is a striking African raptor known for its distinctive behaviors and habitat preferences across grasslands, savannas, and open riverine areas south of the Sahara Desert [De Swardt et al. \(2011\)](#). Its unique plumage, characterized by grey-brown feathers on its back, a pure white chest, and a deep black belly, contributes to its remarkable appearance [Hofmeyr et al. \(2014\)](#). The Secretary Bird's hunting technique is particularly noteworthy. Equipped with long legs and powerful talons, it navigates grasslands by walking or trotting, mimicking the posture of a "secretary at work" as it attentively scans the ground for prey hidden in the vegetation [Portugal et al. \(2016\)](#). Its diet includes insects, reptiles, small mammals, and occasionally larger prey like snakes, showcasing its adaptability and predatory prowess [Portugal et al. \(2016\); Feduccia and Voorhies \(1989\)](#).

When hunting snakes, the Secretary Bird exhibits exceptional intelligence and agility. It uses its height advantage to survey the ground, closely monitoring the movements of snakes. Drawing from years of experience, it predicts the snake's actions and strategically positions itself for an effective strike. This methodical approach includes teasing and provoking the snake, exhausting it before delivering a decisive blow with its sharp talons [Feduccia and Voorhies](#)

(1989); De Swardt et al. (2011). Against larger snakes, which pose a greater threat due to their constricting abilities, the Secretary Bird may lift them off the ground to neutralize their advantage, illustrating its adaptive hunting strategies.

In addition to its hunting prowess, the Secretary Bird employs sophisticated evasion tactics against predators. It utilizes camouflage in suitable environments to blend seamlessly into its surroundings, effectively evading detection. When camouflage is not feasible, the bird resorts to rapid walking or confrontational behavior to deter threats Hofmeyr et al. (2014). These adaptive behaviors reflect the bird's ability to navigate and thrive amidst diverse ecological pressures.

Secretary Bird Optimization Algorithm (SBOA) The Secretary Bird Optimization Algorithm (SBOA), introduced by Youfa Fu and colleagues in 2024, presents a novel metaheuristic approach for solving global optimization problems Fu et al. (2024). Inspired by Secretary Bird's hunting strategies, SBOA mimics the bird's systematic approach to exploration and exploitation in search of optimal solutions.

SBOA offers several advantages over traditional optimization algorithms. It effectively balances global exploration and local exploitation, leveraging its inspiration from Secretary Bird's methodical scanning and decisive striking behaviors Fu et al. (2024). This balance enhances its ability to discover diverse and promising solutions across various domains, including engineering and finance.

The algorithm's ability to predictively adapt, akin to the Secretary Bird's anticipation of prey movements, enhances its effectiveness in complex optimization landscapes Fu et al. (2024). By integrating adaptive mechanisms, SBOA can refine solutions iteratively, optimizing performance metrics and achieving competitive results in benchmark tests.

However, like any algorithm, SBOA has its limitations. Its performance heavily relies on parameter tuning and problem-specific configurations, which can impact its robustness across different optimization tasks Fu et al. (2024). Additionally, while inspired by natural behaviors, the algorithm's applicability and scalability in large-scale optimization scenarios require further exploration and validation.

In conclusion, the SBOA represents a promising advancement in metaheuristic optimization, leveraging nature-inspired strategies to tackle complex global optimization challenges. While demonstrating notable advantages in solution quality and efficiency, ongoing research and refinement is essential to fully harness its potential across diverse applications.

3.2.2. Mathematical Model of the SBOA

The mathematical model of the SBOA emulates the natural behaviors of secretary birds in hunting and evading predators.

Initial Preparation Phase

SBOA is a population-based metaheuristic approach where each secretary bird is a member of the algorithm's population. The position of each secretary bird in the search space represents candidate solutions to the problem at hand. The positions are initialized randomly within the upper (ub) and lower (lb) bounds as described by Equation 4.

$$X_{i,j} = lb_j + r \times (ub_j - lb_j), \quad i = 1, 2, \dots, N, \quad j = 1, 2, \dots, \text{Dim} \quad (4)$$

Where:

- $X_{i,j}$ is the position of the i -th secretary bird in the j -th dimension.
- lb_j and ub_j are the lower and upper bounds of the j -th dimension.
- r is a random number between 0 and 1.

The objective function values of the candidate solutions are evaluated using Equation 5.

$$F_i = F(X_i), \quad i = 1, 2, \dots, N \quad (5)$$

Where:

- F_i is the objective function value of the i -th secretary bird.
- X_i is the position vector of the i -th secretary bird.

Hunting Strategy of Secretary Bird (Exploration Phase)

The hunting behavior of secretary birds is divided into three stages: searching for prey, consuming prey, and attacking prey.

Stage 1: Searching for Prey This stage employs a differential evolution strategy to enhance diversity and global search capabilities. The new position of the i -th secretary bird in this stage is given by Equation 6.

$$X_{\text{new},P1,i,j} = X_{i,j} + (X_{\text{random_1}} - X_{\text{random_2}}) \times R1, \quad t < \frac{1}{3}T \quad (6)$$

Where:

- $X_{\text{random_1}}$ and $X_{\text{random_2}}$ are random candidate solutions.
- $R1$ is a randomly generated array from the interval $[0, 1]$.

The position update decision is made using Equation 7.

$$X_i = \begin{cases} X_{\text{new},P1,i} & \text{if } F_{\text{new},P1,i} < F_i \\ X_i & \text{otherwise} \end{cases} \quad (7)$$

Stage 2: Consuming Prey This stage introduces Brownian motion to simulate the random movement of the secretary bird. The new position is updated using Equation 8.

$$X_{\text{new},P1,i,j} = X_{\text{best}} + \exp\left(\left(\frac{t}{T}\right)^4\right) \times (RB - 0.5) \times (X_{\text{best}} - X_{i,j}), \quad \frac{1}{3}T < t < \frac{2}{3}T \quad (8)$$

Where:

- RB is a random array from a standard normal distribution.
- X_{best} is the current best position.

The position update decision follows Equation 9.

$$X_i = \begin{cases} X_{\text{new},P1,i} & \text{if } F_{\text{new},P1,i} < F_i \\ X_i & \text{otherwise} \end{cases} \quad (9)$$

Stage 3: Attacking Prey This stage uses the Levy flight strategy for enhanced exploration. The new position is updated using Equation 10.

$$X_{\text{new},P1,i,j} = X_{\text{best}} + \left((1 - \frac{t}{T}) \times 2 \times \frac{t}{T}\right) \times X_{i,j} \times RL, \quad t > \frac{2}{3}T \quad (10)$$

Where:

- RL represents the weighted Levy flight.

The position update decision is made using Equation 11.

$$X_i = \begin{cases} X_{\text{new},P1,i} & \text{if } F_{\text{new},P1,i} < F_i \\ X_i & \text{otherwise} \end{cases} \quad (11)$$

The Levy flight distribution function is defined by Equation 12.

$$\text{Levy}(Dim) = s \times \frac{u \times \sigma}{|v|^{1/\eta}} \quad (12)$$

Where:

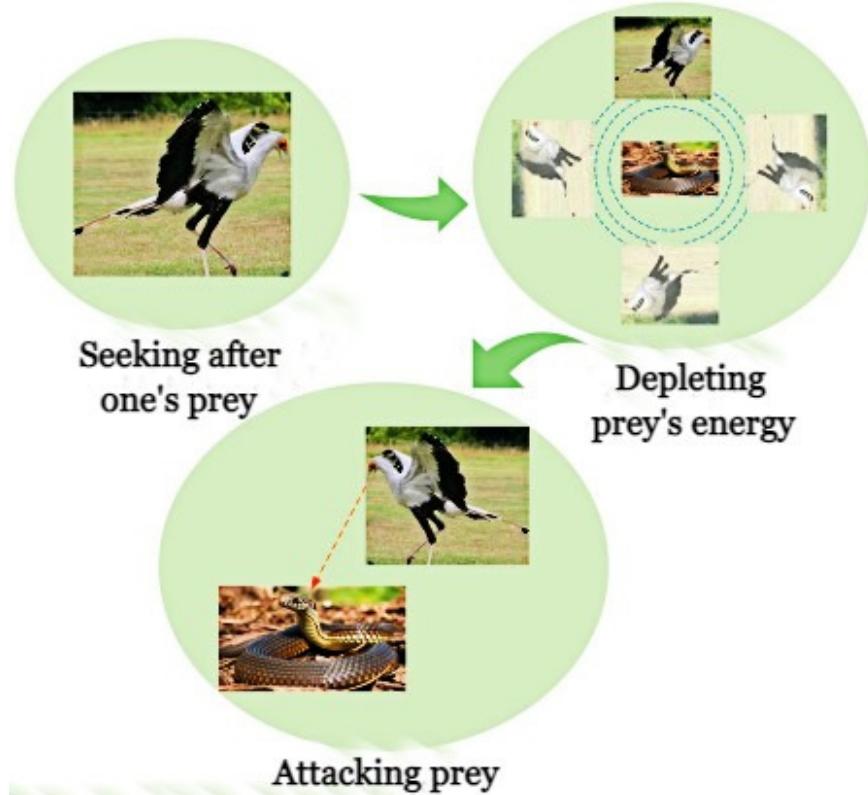
- s is a constant (0.01).
- u and v are random numbers in $[0, 1]$.
- η is a constant (1.5).

The parameter σ is calculated as follows:

$$\sigma = \left(\frac{\Gamma(1 + \eta) \times \sin(\pi\eta/2)}{\Gamma((1 + \eta)/2) \times \eta \times 2^{(\eta-1)/2}} \right)^{1/\eta} \quad (13)$$

The hunting behavior of secretary birds when feeding on snakes generally consists of three stages: searching for prey, attacking prey, and consuming prey. This process is depicted in Figure 2

Fig. 2. The predatory behavior of secretary birds



Escape Strategy of Secretary Bird (Exploitation Stage)

The escape strategy includes two behaviors: camouflage and fleeing. The dynamic perturbation factor aids in balancing exploration and exploitation. The update rule for escape strategies is given by Equation 14.

$$X_{\text{new},P2,i,j} = \begin{cases} X_{\text{best}} + (2 \times RB - 1) \times (1 - \frac{t}{T})^2 \times X_{i,j} & \text{if } r < 0.5 \\ X_{i,j} + R2 \times (X_{\text{random}} - K \times X_{i,j}) & \text{otherwise} \end{cases} \quad (14)$$

Where:

- $R2$ is a random array from a normal distribution.

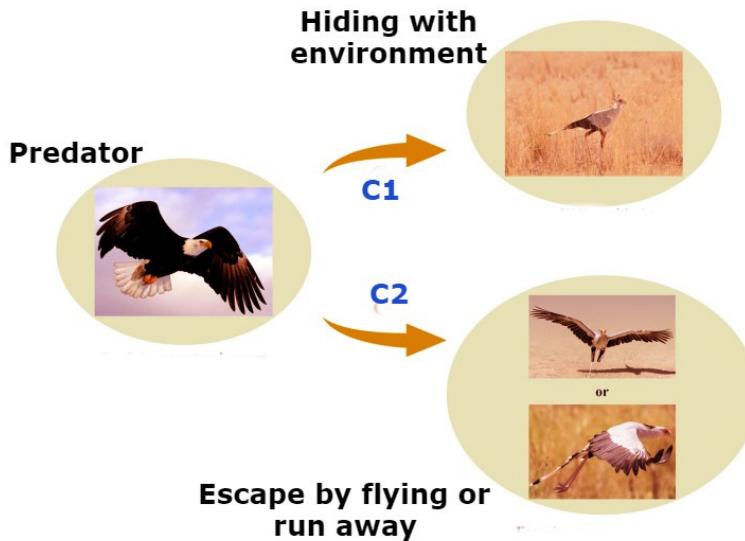
- K is a random integer (1 or 2).

The position update decision is given by Equation 15.

$$X_i = \begin{cases} X_{\text{new},P2,i} & \text{if } F_{\text{new},P2,i} < F_i \\ X_i & \text{otherwise} \end{cases} \quad (15)$$

Secretary birds may blend into their environment using colors or structures, making it difficult for predators to spot them. Their evasion behaviors when faced with threats are shown in Figure 3. The design of the SBOA assumes that one of the following two conditions occurs with equal probability; C1 for Camouflaging within the environment and C2; for Flying or running away.

Fig. 3. The secretary bird's escape behavior



3.3. Enhancement techniques for SBOA

3.3.1. Opposition-Based Learning

Opposition-based learning (OBL) (Tizhoosh, 2005) is a valuable technique for preventing stagnation in competitive solutions (Aarts et al., 2003). This concept enhances the exploitation aspect of the search mechanism. In meta-heuristic algorithms, convergence tends to occur quickly if the initial solutions are near the optimal position; otherwise, convergence is slower. OBL improves outcomes by exploring search regions close to the global optimum. It operates by simultaneously searching in two directions: one defined by the current solution and the other by its opposite. The OBL strategy selects the best direction from all solutions (Houssein et al., 2022).

- **Opposite Number:** The notion of opposite numbers is fundamental to OBL. For a real number $Y_0 \in [u, p]$, the opposite number is given by Eq. (16) (Emam et al., 2023b):

$$\bar{Y}_0 = u + p - Y_0. \quad (16)$$

In an M-dimensional space, the opposite number is calculated using Eq. (17) and Eq. 18:

$$Y = [Y_1, Y_2, Y_3, \dots, Y_M] \quad (17)$$

Algorithm 1 Pseudo code of SBOA

```

1: Initialize population of secretary birds with positions  $X_{i,j}$  using Equation 4
2: Evaluate initial objective function values  $F_i$  using Equation 5
3: for each iteration  $t = 1$  to  $T$  do
4:   for each secretary bird  $i$  do
5:     if  $t < \frac{1}{3}T$  then
6:       Update position using Equation 6
7:       Update position using Equation 7
8:     else if  $\frac{1}{3}T < t < \frac{2}{3}T$  then
9:       Update position using Equation 8
10:      Update position using Equation 9
11:    else
12:      Update position using Equation 10
13:      Update position using Equation 11
14:    end if
15:    Update position using Equation 14
16:    Update position using Equation 15
17:  end for
18: end for
  
```

$$\bar{Y} = [\bar{Y}_1, \bar{Y}_2, \bar{Y}_3, \dots, \bar{Y}_M] \quad (18)$$

Each component of \bar{Y} is defined by Eq. (19):

$$\bar{Y}_k = u_q + p_q - Y_q \quad \text{where } q = 1, 2, 3, \dots, M \quad (19)$$

- **Opposition-Based Optimization:** In this optimization approach, the opposite value \bar{Y}_0 is evaluated against the current solution Y_0 using the fitness function. If $f_t(Y_0)$ is superior to $f_t(\bar{Y}_0)$, then Y_0 remains unchanged; otherwise, Y_0 is replaced by \bar{Y}_0 . Consequently, the solutions are updated to reflect the best value between Y and \bar{Y} (Emam et al., 2023b).

3.3.2. Orthogonal Learning (OL)

Orthogonal Learning (OL) is a technique extensively used to enhance the search for optimal solutions, striking a balance between exploration and exploitation phases (Zhang et al., 2020). The orthogonal experimental design (OED) method is utilized to develop an OL strategy, crafting efficient agents that guide the population towards the global optimal solution (Gao et al., 2013). The OED determines the best combination of factor levels using a small sample of experiments, which provides new solutions that steer the search in the optimal direction. The OL strategy is implemented in two primary stages:

Orthogonal Array (OA): The first stage involves generating a predefined table known as the orthogonal array (OA). This table consists of a series of distinct numbers typically represented as $L_M(L^Q)$, where $M = 2 \cdot \log_2(D+1)$. The generated OA has the following properties: L levels per factor and Q factors. For instance, the OA for a three-dimensional Sphere function $O_{array}(2^3)$ is illustrated below.

Combination	x1(Level)	x2(Level)	x3(Level)	Fitness
C1	1(1)	2(1)	3(1)	$f(C1) = 14$
C2	1(1)	1(2)	4(2)	$f(C2) = 18$
C3	2(2)	2(1)	4(2)	$f(C3) = 24$
C4	2(2)	1(2)	3(1)	$f(C4) = 14$

In this example, the OA comprises three columns, indicating its application for problems with up to three factors, each having two levels.

Factor Analysis (FA): The second stage involves factor analysis, which evaluates the effect of each level on each factor based on the experimental results of all M combinations of the OA. This effect is calculated using the following equation:

$$W_{q,l} = \sum_{m=1}^M f(C_m) \cdot E_{m,q,l} \quad (20)$$

Here, $W_{q,l}$ represents the effect of the l^{th} level on the q^{th} factor, where $m = 1, 2, 3, \dots, M$, $q = 1, 2, 3, \dots, Q$, and $l = 1, 2, 3, \dots, L$. The term $f(C_m)$ denotes the fitness of the m^{th} combination in the OA. The variable $E_{m,q,l}$ is set to 1 if the l^{th} level is used for the q^{th} factor in the m^{th} combination, and 0 otherwise. Using Eq. (20), the impact of each level on each factor, as shown in Table 3, is quickly determined. By comparing these effects, the optimal combination of levels is identified.

Table 3 Orthogonal experimental design for a three-dimensional Sphere function.

Level	Factor Analysis		
L1	$f(C1)+f(C2)=32$	$f(C1)+f(C3)=38$	$f(C1)+f(C4)=28$
L2	$f(C3)+f(C4)=38$	$f(C2)+f(C4)=32$	$f(C2)+f(C3)=42$
Best Level	x1(1)	x2(2)	x3(1)
OED Result	1	1	3
			$f_{min}=11$

Overall, the OL strategy can be expressed as Eq. (21):

$$X_n^m = X_{n_{best}}^m \oplus X_n^m \quad (21)$$

In this equation, the symbol \oplus represents the OL operation. In this study, an agent X_n^m is selected from the population, and the best-scoring agent $X_{n_{best}}^m$ is combined with it to generate a new, efficient solution.

3.4. Enhanced Secretary Bird Optimization Algorithm (mSBOA)

This subsection delves into the specifics of the enhanced Secretary Bird Optimization Algorithm (mSBOA), highlighting the enhancements made to augment its exploitation capabilities and expedite the exploration phase. Furthermore, it addresses strategies implemented to mitigate the risks of getting trapped in local optima and premature convergence.

The mSBOA incorporates two advanced schemes to achieve these improvements:

- Opposition-based Learning:** This technique enhances the diversity of the population by considering both the current solution and its opposite, thus increasing the chances of escaping local optima and improving the overall search efficiency.
- Orthogonal Learning (OL):** is a technique extensively used to enhance the search for optimal solutions, striking a balance between exploration and exploitation phases

By using the OL technique, SBOA solutions are improved, searching capacity is increased, and effective solutions are generated, which should ultimately lead to the premium solution. The OL strategy (illustrated in Subsect 3.3.2) is also employed based on Eq. (21) to generate another population of solutions, namely X_n . Then, the best fittest solutions from X_{if}^{new} are selected and saved as a current solution instead of X_{best} .

3.5. Initialization phase

According to the SBOA algorithm, the mSBOA algorithm begins by developing an initial population (N_n); each population has a dimension (*Dim*) in the search space limited by the lower and upper boundaries (l_b and u_b). The positions of secretary birds are defined according to l_b and u_b as shown in Eq. 22. In addition, the maximum number of iterations (*Maxitr*) and the selection probabilities ($p1, p2$) are considered.

$$X b_i = l_b + \text{rand} \times (u_b - l_b); \quad i = 1, 2, \dots, N_n \quad (22)$$

Then, the mSBOA's diversity is improved in the search process using the OBL strategy during the initialization phase to enhance the search operation, as shown in Eq. 23:

$$Opp_s = l_b + u_b - x_b, \quad b \in \{1, 2, \dots, N_n\} \quad (23)$$

where Opp_s is a vector obtained by performing the OBL. In addition to OBL, Brownian motion is incorporated to introduce random perturbations that aid in fine-tuning the solutions, balancing exploration and exploitation.

The combination of these advanced schemes in mSBOA significantly enhances its capability to avoid local optima and achieve a more robust global search, ensuring a more efficient optimization process.

3.6. Fitness evaluation phase

In this phase, each proposed solution's fitness score F is evaluated, and the best solution throughout the current iteration X_{best} is identified.

3.7. Updating phase

The proposed mSBOA algorithm initiates its position updating process by employing the Searching for Prey phase, updating the solution according to Equations 6 and 7. Subsequently, Brownian motion is introduced to simulate the random movement of the secretary bird, with the new position updated using Equation 8. The decision for position update follows Equation 9.

Next, the Attacking Prey phase is applied. This stage employs the Levy flight strategy to enhance exploration, updating the new position using Equation 10. The decision for the position update is made using Equation 11.

Ultimately, after each update, the best agent and fitness score are replaced and recorded using the positive greedy selection method. The general concept involves comparing an agent's updated fitness score with the global optimum; if the updated score surpasses the existing global optimum, the agent is recorded as the new optimum, and the optimum fitness score is updated accordingly.

3.8. OL phase

The OL strategy is performed based on Eq. (21) to generate new solutions based on the OL strategy, namely X_n . Then, each solution from X^{new} is evaluated to obtain its fitness $F(X_n)$. Finally, the best-fitness positions from X_n are then determined and saved in X as the current solution. Hence, when the quality of the new solutions is iteratively improved, the balance between the diversification and intensification stages can be achieved. Therefore, getting trapped in the local optimal positions can be subsequently avoided. The pseudo-code is depicted in Algorithm 2

4. Extensive analysis and preprocessing of SCINE dataset

The Skin Condition Image Network (SCIN) dataset offers a diverse and representative collection of skin condition images, bridging important gaps for AI development, medical research, and equitable healthcare tools. Various new techniques in data science are implemented on the unstructured SCIN dataset. Figure 4 shows these techniques:

4.1. Defining SCIN dataset

- Dataset composition

The SCIN dataset currently contains over 10,000 images of skin, nail, or hair conditions, directly contributed by individuals experiencing them. All contributions were made voluntarily with informed consent by individuals in the US, under an institutional review board-approved study. To provide context for retrospective dermatologist labeling, contributors were asked to take images both close-up and from slightly further away. They were given the option to self-report demographic information and tanning propensity (self-reported Fitzpatrick Skin Type, i.e., sFST), and to describe the texture, duration, and symptoms related to their concern. One to three dermatologists labeled each contribution with up to five dermatology conditions, along with a confidence score for each label. The SCIN dataset contains these individual labels, as well as an aggregated and weighted

Algorithm 2 Pseudo-code of mSBOA

```
1: Initialize the population of secretary birds with positions  $X_{i,j}$  using Equation 4
2: Evaluate initial objective function values  $F_i$  using Equation 5
3: for  $i = 1$  to  $N_n$  do
4:    $X_i = l_{bi} + \text{rand} \times (u_{bi} - l_{bi})$ 
5:   Perform Opposite-Based Learning (OBL) on the initial population using Equation 23 and save the result in  $Opp_s$ 
6:   Evaluate  $X_i$  using the fitness function and store the results in  $fit_i$ 
7:   Compute the fitness value
8:   if  $fit_i < fit_{Opp_i}$  then
9:      $X_i = Opp_i$ 
10:    end if
11: end for
12: for each iteration  $t = 1$  to  $T$  do
13:   for each secretary bird  $i$  do
14:     if  $t < \frac{1}{3}T$  then
15:       Update position using Equation 6
16:       Update position using Equation 7
17:     else if  $\frac{1}{3}T < t < \frac{2}{3}T$  then
18:       Update position using Equation 8
19:       Update position using Equation 9
20:     else
21:       Update position using Equation 10
22:       Update position using Equation 11
23:     end if
24:     Update position using Equation 14
25:   Opposition Learning (OL) phase
26:   Apply OL phase
27:   Update position using Equation 15
28: end for
29: end for
```

differential diagnosis derived from them that could be useful for model testing or training. These labels were assigned retrospectively and are not equivalent to a clinical diagnosis, but they allow us to compare the distribution of dermatology conditions in the SCIN dataset with existing datasets.

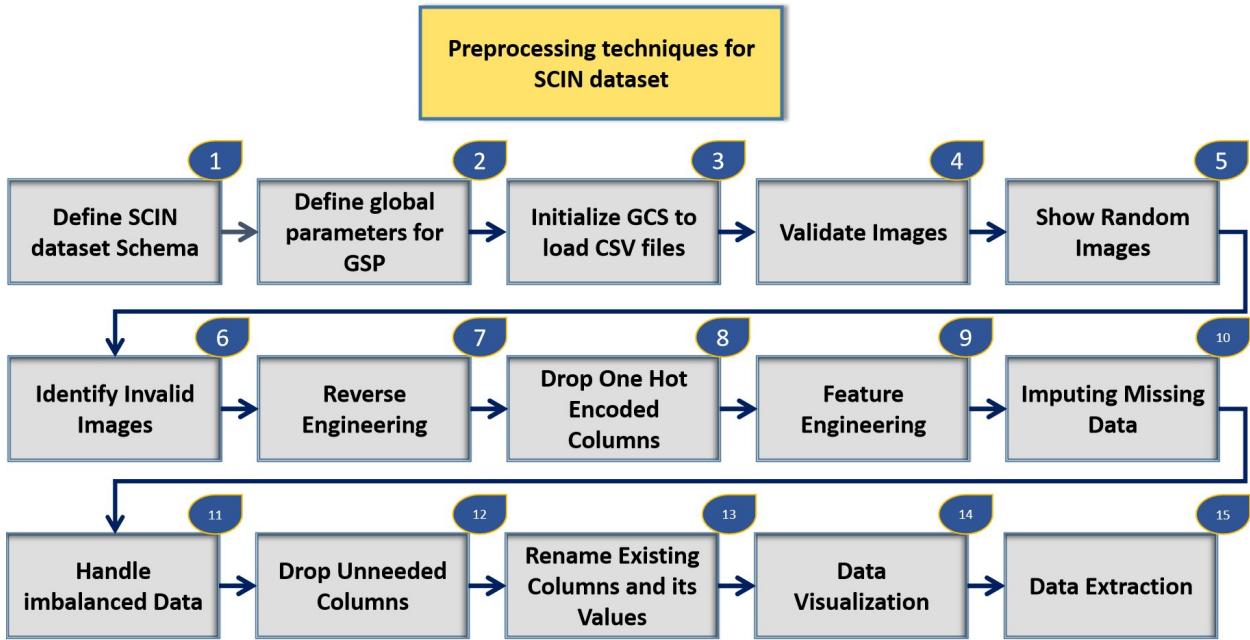
- Fitzpatrick scale

Before working with the SCIN dataset you need to know what is Fitzpatrick scale. The Fitzpatrick scale, also known as the Fitzpatrick skin type classification, is a system for classifying human skin color based on the skin's response to ultraviolet (UV) radiation exposure. Developed by dermatologist Thomas B. Fitzpatrick in 1975, this scale helps to categorize skin types and predict the risk of sunburn and skin cancer. It is widely used in dermatology and skincare to tailor treatments and preventive measures based on skin type. preventive measures based on skin type.

- Monk Skin Tone Scale

The Monk Skin Tone Scale, also known as the Monk skin type classification, is a system developed for classifying human skin tones. It aims to provide a simple and practical way to categorize skin types based on their susceptibility to sunburn and skin cancer, similar to the Fitzpatrick scale. The Monk Skin Tone Scale typically includes a range of skin types, often categorized into several groups based on their characteristics, such as Light Skin Tones, Medium Skin Tones, Dark Skin Tones, and Very Dark Skin Tones.

Fig. 4. Data science techniques applied on SCIN dataset in preprocessing stage



4.2. Dataset schema

- Dataset on Google Cloud Storage

The data is stored in Google Cloud Storage and to access it click [HERE](#). Check out the load notebook from [HERE](#) for a quick review of how to access the dataset and the Dataset Documentation for an overview of its schema from [HERE](#). Please note: This dataset contains images of medical conditions, some of which may be sensitive and/or graphic in nature.

- Dataset Description

1- scin_cases.csv

- case_id: Identifier for the case.
- source: Value is "SCIN".
- release: Identifier for the release of the case, formatted as "major.minor.patch".
- year: The year closest to the bulk of the data released.
- age_group: Mapped age ranges: AGE_18_TO_29, AGE_30_TO_39, etc.
- sex_at_birth: User-reported sex at birth: FEMALE, MALE, OTHER_OR_UNSPECIFIED.
- fitzpatrick_skin_type: User-reported skin type: FST1-6, NONE_SELECTED.
- race_ethnicity_*: User-reported race and/or ethnicity demographic.
- textures_*: User-reported skin condition textures.
- body_parts_*: User-reported affected body parts.
- condition_symptoms_*: User-reported symptoms related to the skin condition.
- other_symptoms_*: User-reported additional symptoms.
- related_category: User-reported related categories.
- condition_duration: User-reported duration of the skin condition.
- image_path: Path to the image storage location.

- image_shot_type: Enum indicating image shot type.

2-skin_labels.csv

- case_id: Same as skin_cases.csv.
- dermatologist_gradable_for_skin_condition: Label indicating if skin condition can be determined.
- dermatologist_skin_condition_label_name: List of condition names derived from dermatologist labels.
- dermatologist_skin_condition_confidence: Confidence scores for dermatologist labels.
- weighted_skin_condition_label: Final differential label generated from dermatologist labels.
- dermatologist_gradable_for_fitzpatrick_skin_type: Label indicating if Fitzpatrick skin type can be estimated.
- dermatologist_fitzpatrick_skin_type_label: Dermatologist's estimated Fitzpatrick skin type.
- gradable_for_monk_skin_tone_india: Label indicating if Monk skin tone label can be determined in India.
- gradable_for_monk_skin_tone_us: Label indicating if Monk skin tone label can be determined in the US.
- monk_skin_tone_label_india: Monk skin tone label value in India.
- monk_skin_tone_label_us: Monk skin tone label value in the US.

- Accessing the Dataset from Google Cloud Storage (GCS)

- **Defines Global Parameters for Google Cloud Platform (GCP)**

To access the dataset from Google Cloud Storage you need to define a class called Globals with several parameters related to a Google Cloud Platform (GCP) project and Google Cloud Storage (GCS) bucket. These parameters include the GCP project name, GCS bucket name, paths to CSV files containing metadata and labels, and the directory containing images within the bucket. The class also initializes some variables like a GCS storage client, bucket object, and DataFrames for the loaded CSV files. Finally, it prints out the values of some of the parameters defined in the class. Overall, this code sets up global parameters and configurations for a data science or machine learning project that involves accessing data stored in a GCS bucket within a GCP project.

- Create a DataFrame (DF)

After initializing a GCS client and bucket, loads ‘metadata’ and ‘labels’ CSV files from the bucket into pandas DataFrames, and merges them based on ‘case_id’, finally printing the length of the merged DataFrame.

4.3. Other preprocessing steps

- Display a random Images

This step defines functions to display images associated with a case ID and optionally print condition labels. It uses the Pillow library to handle images and Matplotlib to display them. The display_image function loads an image from Google Cloud Storage (GCS) based on a provided image path and displays it using Matplotlib. The display_images_for_case function selects a random case from a df based on the provided case ID (if any) or selects a random case if none is provided. It then retrieves the image paths associated with that case and displays each image using the display_image function. Additionally, it optionally prints condition labels associated with the case. The code is designed to work with a df that contains information about cases, including their IDs, image paths, and condition labels. It seems to be part of a larger project related to image classification or analysis, where cases are associated with images and corresponding condition labels.

- Identify Invalid Images

This step defines a function is_valid_image_path to check if an image path is valid by attempting to open the image using Pillow. It then applies this function to each image path column in the DataFrame Globals.cases_and_labels_df, adding a new column for each image path column to indicate whether the image path is valid or not. Finally, it prints the first few rows of the df with the validity columns. The missing image path is: "dataset/images/-2243186711511406658.png

- Reverse Engineering

This step defines a Python script that reverses one-hot encoding by mapping binary encoded columns back to their original categorical values, iterating over specified groups, and creating new df columns for each group's categorical values. It updates these columns with category names where the corresponding one-hot encoded column is 'YES', effectively restoring the original categorical values.

- Dropping one hot encoded column

After applying reverse engineering on the one-hot encoded columns to obtain a normal categorical column, all the encoded columns are deleted because they don't provide meaningful information. The dropping of encoded is performed by initializing a list that is initially empty. Then, create a for loop that iterates over all the columns in the df to check if the column name starts with any prefix in the one_hot_groups dictionary. If it does, it is added to the list. Finally, all the columns in this list are dropped.

- Feature engineering process

In this phase, we calculate the most common value for each row across multiple sets of columns related to dermatologist gradings for skin conditions and Fitzpatrick skin types in a data frame. It defines a function to calculate the mode for each row and applies it to each set of columns using the apply function along the rows (axis=1) of the DataFrame, generating new columns with the most common values for each set of columns.

- Impute Missing Values & Fix Unbalanced Data

First, the categorical labels are decoded using LabelEncoder to convert them into numerical values. After this, the missing values are imputed in the encoded column with the most frequent value using SimpleImputer. Then, the imputed numerical values are decoded back to their original categorical labels using LabelEncoder. Finally, the data frame is updated with the imputed categorical values in a new column.

- Dropping Unneeded Columns

Because they either contain mostly null values or are not relevant to your analysis. The indices correspond to the following columns: 'case_id', source, release, year: These seem to be identifiers or metadata that are not necessary for your analysis. 'race_ethnicity_two_or_more_after_mitigation': Contains mostly null values. 'dermatologist_gradable_for_skin_condition_': These columns have many null values and might not be useful for your analysis. 'dermatologist_skin_condition_on_label_name', 'dermatologist_skin_condition_confidence, weighted_skin_condition_label', 'dermatologist_fitzpatrick_skin_type_label_': Likely not relevant or redundant for your analysis. 'gradable_for_monk_skin_tone_,' monk_skin_tone_label_': These columns seem specific to certain analyses and are not relevant to your current task. By dropping these columns, you're likely simplifying your dataset to focus on the most relevant features for your analysis, which can improve model performance and reduce computational overhead.

- Renaming columns and their values

In this step, the unclear column names are renamed to make them easier to recognize. Also, values in some columns are renamed to clarify their meaning for the reader. For example, rename columns to more descriptive names, e.g., 'fitzpatrick_skin_type' to 'fitzpatrick_scale', and 'image_1_path' to 'image_name'. Split and clean the 'textures' column to remove the prefix and convert values to lowercase. Replace underscores in the 'textures' column with spaces for readability.

- Data Visualization

In this step includes creating visuals to gain insights, such as detecting data bias, identifying missing entries, and determining if any changes are needed in the data frame.

- Data Extraction

- Creating a function to delete records from the DataFrame that don't have corresponding images in the image folder, based on the image_name column.
- Validate that each record has an image in the images folder.

- Preprocess the image data.
- Get 15 valid sample records.
- Create another DataFrame called Val_samples to store these samples for further research.

5. Experiments: specifications, results and discussion

5.1. Parameters setup

Table 4 outlines the parameter configurations for the compared algorithms. Following the findings of Arcuri and Briand ([Arcuri and Fraser, 2013](#)), which indicate that default parameter values suffice for fair algorithm comparisons, we adopt default values to ensure impartiality. Using default settings mitigates potential bias from algorithm-specific tuning, providing a more objective performance evaluation. To enhance the robustness and reliability of the comparison, each simulation is conducted independently 30 times. This repetition accounts for variations in performance due to random initialization or stochastic processes, leading to a more precise evaluation of the algorithms' efficiency and effectiveness.

Table 4 Parameter setting of compared algorithms

Algorithm	Parameter setting
Common Settings	Population size: $N = 30$ Maximum fitness evaluation: $\max_{FE} = 30.000$ Dimensions $Dim = 10$ Number of runs : 30
WOA	α decreases from 2 to 0 a_2 decreases from -1 to -2
GWO	a decreases from 2 to 0
HHO	$\beta = 1.5$
HBA	$\beta = 3$ $W = 0.8$ $p = 0.03$
BWO	$B_F = (0, 0.5)$
RSA	default
SBOA and mSBOA	default

5.2. First series experiments: Implementation of mSBOA for global optimization using CEC 2022 engineering benchmark functions

5.2.1. CEC 2022 Benchmark Test Functions

The CEC 2022 benchmark functions, among the latest and most challenging, are used to evaluate the effectiveness of the proposed mSBOA algorithm. This test suite includes 12 benchmark functions extensively utilized in optimization research. Recognized as a contemporary problem set for optimization algorithm assessment, CEC 2022 comprises various functions designed to address different optimization challenges. Table 5 provides a detailed overview of the characteristics of the CEC 2022 functions, while Figure 5 presents their two-dimensional visualizations. In addition to standard benchmark functions, the CEC 2022 suite features hybrid and composite functions. Hybrid functions split variables into subcomponents, applying different base functions to each, which mirrors real-world optimization problems where subcomponents exhibit distinct properties. Composite functions, on the other hand, create a more complex optimization landscape by combining shifted and rotated versions of various base functions. This method captures the intricacies of real-world problems and ensures continuity around both global and local optima. By utilizing the CEC 2022 test suite, which includes a diverse range of benchmark functions, hybrid, and composite performance of the mSBOA algorithm can be comprehensively evaluated. This thorough assessment demonstrates the algorithm's effectiveness in addressing various optimization challenges.

Table 5 CEC 2022 benchmarks.

Function No.	Description	Range	F_{min}
F1	Shifted and full Rotated Zakharov Function	[-100, 100]	300
F2	Shifted and full Rotated Rosenbrock's Function	[-100, 100]	400
F3	Shifted and full Rotated Expanded Schaffer's f6 Function	[-100, 100]	600
F4	Shifted and full Rotated Non-Continuous Rastrigin's Function	[-100, 100]	800
F5	Shifted and full Rotated Levy Function	[-100, 100]	900
F6	Hybrid Function 1 (N = 3)	[-100, 100]	1800
F7	Hybrid Function 2 (N = 6)	[-100, 100]	2000
F8	Hybrid Function 3 (N = 5)	[-100, 100]	2200
F9	Composition Function 1 (N = 5)	[-100, 100]	2300
F10	Composition Function 2 (N = 4)	[-100, 100]	2400
F11	Composition Function 3 (N = 5)	[-100, 100]	2600
F12	Composition Function 4 (N = 6)	[-100, 100]	2700

5.2.2. Performance measurements of mSBOA on CEC'2022 functions

The significance of the mSBOA algorithm is assessed by a set of performance measures. These measures have the subsequent definitions:

- **Statistical Mean:** The statistical mean is the average of the cost values from multiple runs, calculated using Equation (24).

$$Mean = \frac{1}{R_n} \sum_{j=1}^{R_n} Fitt_b^i \quad (24)$$

- **Worst Value:** The worst value is the highest fitness value attained by the algorithm across all runs. It is determined using Equation (25).

$$WORST = \max_{1 \leq j \leq R_n} Fitt_b^i \quad (25)$$

- **Best value:** The best value of the algorithm is defined as the lowest fitness value, which is determined by averaging the results from all runs. This is calculated using Eq. (26).

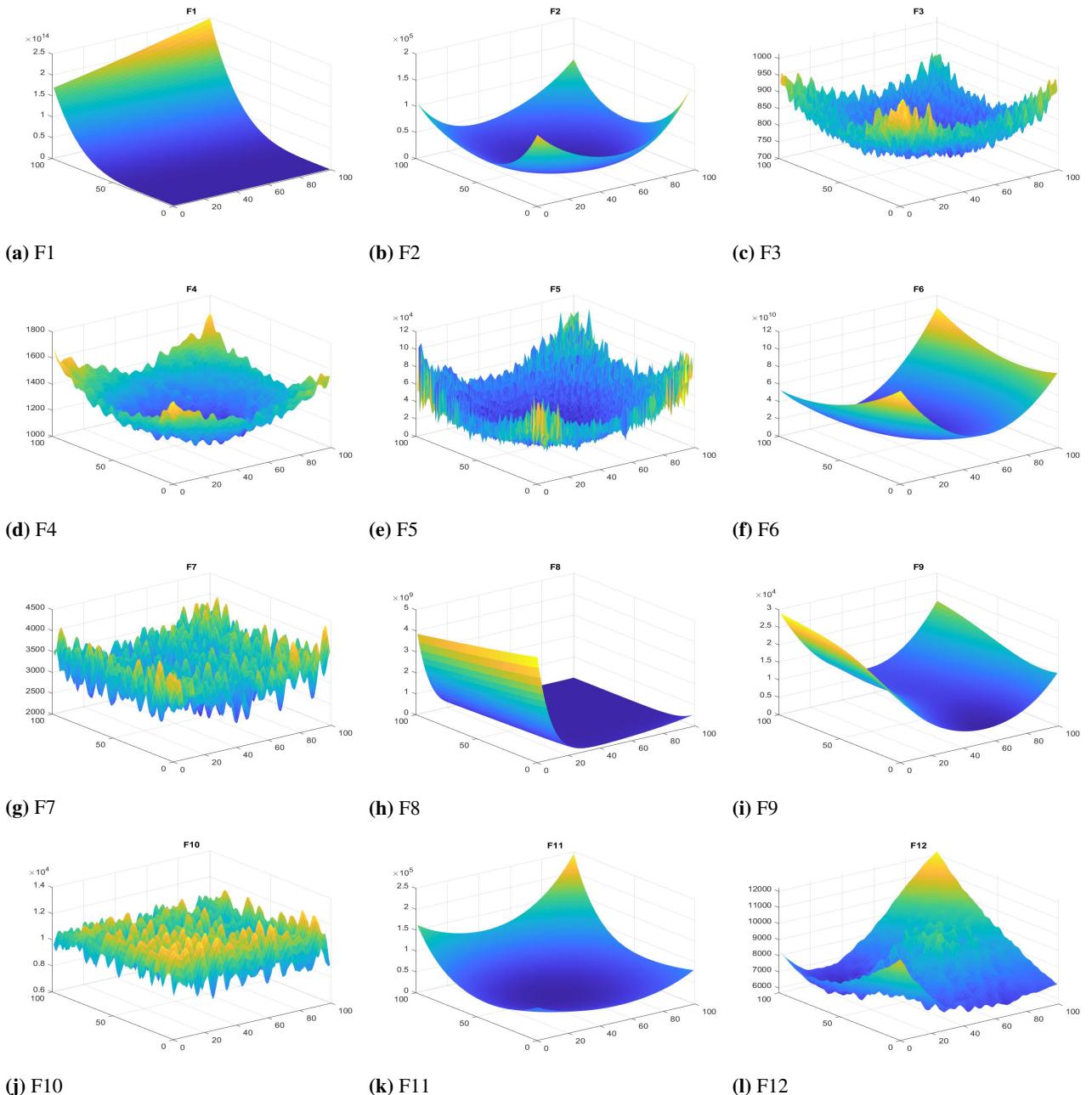
$$BEST = \min_{1 \leq j \leq R_n} Fitt_b^i \quad (26)$$

- **Standard Deviation (STD):** The standard deviation quantifies the spread of the cost values obtained from multiple runs. It is calculated using Eq. (27).

$$STD = \sqrt{\frac{1}{R_n - 1} \sum_{j=1}^{R_n} (Fitt_b^i - Mean)^2} \quad (27)$$

where the number of runs overall is denoted by R_n .

Fig. 5. The 2D of the CEC 2022 functions.



5.2.3. Statistical Results Analysis

This subsection offers a detailed comparison and analysis of the performance of mSBOA against SBOA and several well-known algorithms, including WOA, GWO, HHO, HBA, BWO, and RSA. Table 6 presents the mean, STD, best, and worst values for mSBOA and the other algorithms across 12 test functions with a dimension of 10. Alongside the Friedman rank, these statistical metrics provide a thorough evaluation of the algorithm's performance, facilitating a comprehensive comparison with other optimization techniques.

As illustrated in Table 6, the performance of each algorithm across the 12 benchmark functions varies significantly. For the Unimodal F1 test function, the best performance was obtained by mSBOA with a mean value of 3.069E+02, which is significantly lower than the other algorithms, demonstrating its superior optimization capability. The worst

performance for this function was observed in HBA, with a mean value of 1.075E+07, indicating a substantial gap in efficiency compared to mSBOA. In the case of the F2 function, mSBOA also excelled with a mean value of 4.483E+02, closely followed by SBOA with 4.579E+02, while HBA performed the worst with a mean value of 4.416E+03. This consistent performance of mSBOA highlights its robustness in handling different optimization problems. For the F3 function, mSBOA maintained its top position with a mean value of 6.000E+02, indicating its high accuracy and consistency. Other algorithms, like WOA and GWO, showed slightly higher mean values but performed competitively in this scenario. The F4 function results indicate mSBOA's mean value of 8.341E+02 as the best, showcasing its ability to outperform others like HBA and RSA, which had mean values of 1.052E+03 and 9.677E+02, respectively.

In F5, mSBOA again demonstrated its efficiency with a mean value of 9.025E+02, significantly better than WOA and HBA, which had mean values of 3.863E+03 and 7.625E+03, respectively. This highlights mSBOA's capability to manage more complex optimization challenges effectively. For F6, mSBOA showed an outstanding performance with a mean value of 6.032E+03, whereas HBA struggled with a mean value of 3.972E+09, indicating mSBOA's superior optimization potential in more challenging functions. In the F7 function, mSBOA's mean value of 2.033E+03 was the best, with the closest competitor being GWO with 2.092E+03. This reinforces mSBOA's consistent superiority across various functions. The F8 function results were similar, with mSBOA achieving the best mean value of 2.224E+03, while HBA had the worst mean value of 1.557E+04, further demonstrating mSBOA's robustness. In F9, mSBOA and SBOA shared the best mean value of 2.481E+03, showcasing their joint effectiveness in this particular function, while HBA again performed the worst. For F10, mSBOA achieved a mean value of 2.494E+03, outperforming all other algorithms, with WOA and HBA showing significantly higher mean values, indicating less effective optimization. In the F11 function, mSBOA's mean value of 2.900E+03 was the best, while HBA's 1.041E+04 was the worst, highlighting mSBOA's consistent performance. Finally, in F12, mSBOA's mean value of 2.941E+03 was marginally better than SBOA's 2.943E+03, with HBA again showing the worst performance, demonstrating mSBOA's slight edge in optimization.

Overall, mSBOA consistently performed the best across most of the benchmark functions, achieving the highest rank in the Friedman mean rank with a score of 2.11 and an overall rank of 1. Conversely, HBA often showed the least effective performance, with the lowest overall rank of 8. These results underscore mSBOA's superior optimization capabilities and robustness across diverse optimization problems.

5.2.4. Statistical comparison using Boxplots

We used boxplots to illustrate the properties of the data allocation. To understand the data distributions, we employed boxplots with our results in Table 6. A proper graphical technique for showing data distributions in quartiles is a boxplot. The bottom and upper whiskers of the boxplots are the minimum and maximum data points represented by their edges. In the boxplot, the rectangle's center represents the interquartile range, which separates the lower and higher quartiles. A narrow boxplot represents high consistency and agreement among the data. For clarity, the boxplots for F1 through F12 with a solution of $\text{Dim} = 10$ are shown in Figure 6. The boxplots depict the data's dispersion and distribution, facilitating a thorough comprehension of the algorithm's efficacy across the functions.

The boxplots of the suggested algorithm show optimal and lower values for every tested function. Compared to other algorithms' distributions, boxplots are noticeably narrower. This indicates that the proposed approach works incredibly well. According to the results, the suggested approach performs better than other algorithms in most of the test functions. These results imply that the proposed approach is viable for resolving optimization problems. The proposed algorithm's potential in different fields and practical applications can be investigated further.

5.2.5. Convergence behavior analysis

This section analyzes the convergence of mSBOA compared to other algorithms. Figure 7 displays the convergence plots for WOA, GWO, HHO, HBA, BWO, RSA, and the original SBOA towards the proposed mSBOA on the CEC'2020 benchmarks.

5.2.6. Qualitative Metrics Discussion

Tracking the movements of agents during algorithm convergence and search optimization can offer valuable insights. Figure 8 presents the qualitative analysis of the proposed algorithm, illustrating agent behavior within a 2-D function space. The figure includes convergence plots, search histories, and mean fitness histories to provide a comprehensive view of the search process. By analyzing these behaviors, we can better understand the proposed algorithm's efficiency and performance.

Table 6 CEC 2022 simulation of comparison with other algorithms of 30 experiments.

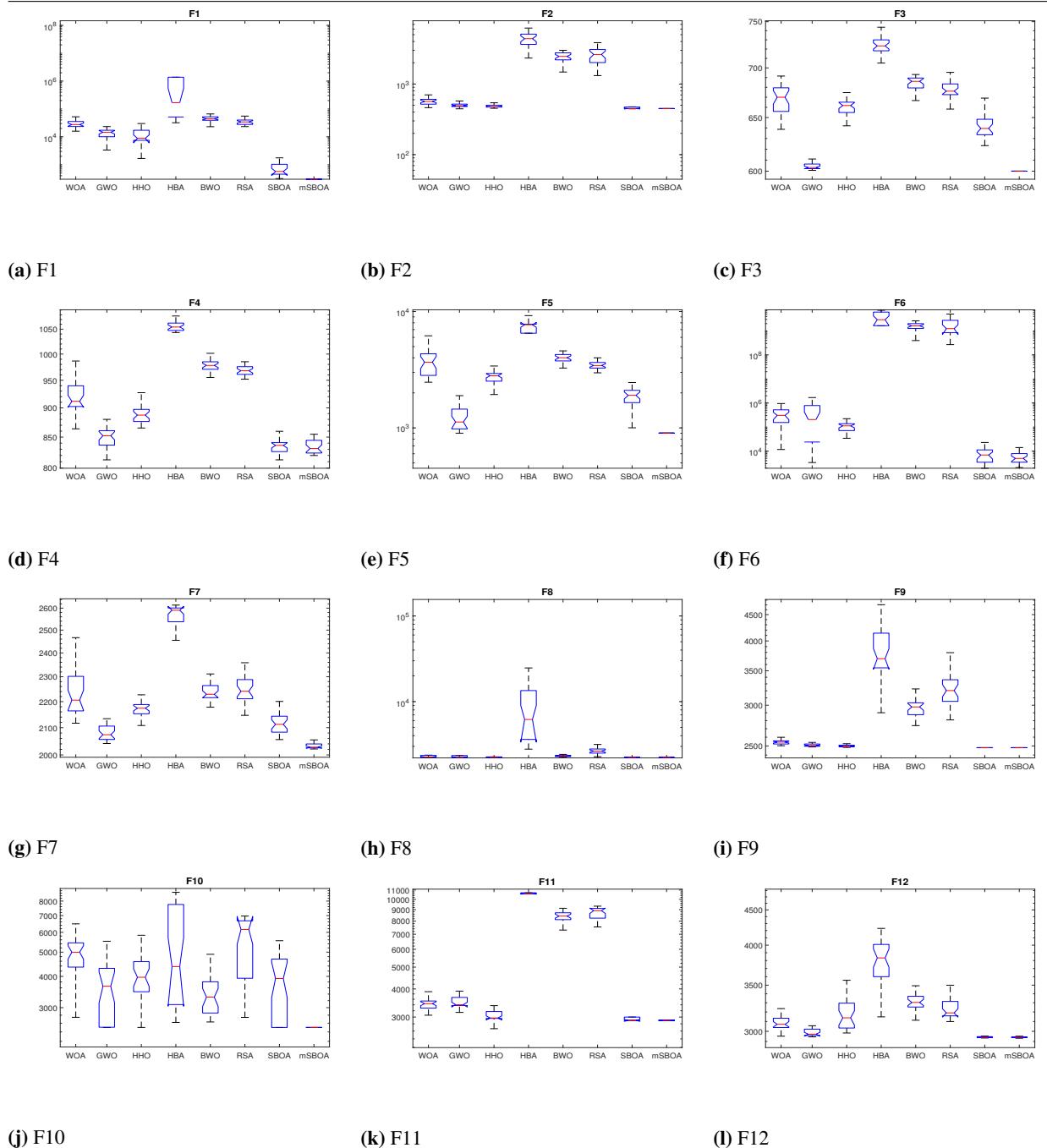
Function	Metric	WOA	GWO	HHO	HBA	BWO	RSA	SBOA	mSBOA
F1	Mean	3.124E+04	1.362E+04	1.176E+04	1.075E+07	4.604E+04	3.767E+04	8.253E+02	3.069E+02
	Std	1.123E+04	5.155E+03	7.305E+03	2.887E+07	1.159E+04	1.532E+04	5.717E+02	2.622E+01
	Best	1.593E+04	3.342E+03	1.675E+03	3.197E+04	2.307E+04	2.326E+04	3.166E+02	3.000E+02
	Worst	6.608E+04	2.334E+04	2.998E+04	1.397E+08	7.253E+04	103947.03	2.264E+03	4.447E+02
F2	Mean	5.670E+02	5.075E+02	4.889E+02	4.416E+03	2.424E+03	2.556E+03	4.579E+02	4.483E+02
	Std	6.372E+01	4.873E+01	2.709E+01	1.188E+03	3.947E+02	6.766E+02	1.165E+01	1.847E+01
	Best	4.624E+02	4.452E+02	4.529E+02	2.329E+03	1.478E+03	1.316E+03	4.449E+02	4.002E+02
	Worst	7.019E+02	6.554E+02	5.608E+02	7.509E+03	3.002E+03	3.852E+03	4.753E+02	4.706E+02
F3	Mean	6.682E+02	6.045E+02	6.600E+02	7.226E+02	6.840E+02	6.771E+02	6.420E+02	6.000E+02
	Std	1.486E+01	3.209E+00	8.721E+00	1.164E+01	7.488E+00	8.484E+00	1.052E+01	2.369E-02
	Best	6.387E+02	6.007E+02	6.395E+02	6.899E+02	6.667E+02	6.583E+02	6.233E+02	6.000E+02
	Worst	6.916E+02	6.144E+02	6.747E+02	7.439E+02	6.931E+02	6.954E+02	6.691E+02	6.001E+02
F4	Mean	9.194E+02	8.573E+02	8.883E+02	1.052E+03	9.781E+02	9.677E+02	8.355E+02	8.341E+02
	Std	3.115E+01	3.019E+01	1.559E+01	1.848E+01	1.107E+01	9.567E+00	1.067E+01	1.081E+01
	Best	8.639E+02	8.129E+02	8.653E+02	1.009E+03	9.553E+02	9.520E+02	8.129E+02	8.199E+02
	Worst	9.865E+02	9.398E+02	9.296E+02	1.077E+03	1.001E+03	9.851E+02	8.597E+02	8.547E+02
F5	Mean	3.863E+03	1.215E+03	2.739E+03	7.625E+03	3.967E+03	3.423E+03	1.842E+03	9.025E+02
	Std	1.266E+03	2.640E+02	3.207E+02	8.781E+02	3.979E+02	2.997E+02	3.597E+02	3.007E+00
	Best	2.470E+03	9.008E+02	1.935E+03	6.514E+03	2.910E+03	2.554E+03	1.002E+03	9.002E+02
	Worst	7.071E+03	1.893E+03	3.403E+03	9.904E+03	4.591E+03	4.001E+03	2.454E+03	9.132E+02
F6	Mean	4.833E+05	1.452E+06	1.195E+05	3.972E+09	1.554E+09	1.836E+09	9.061E+03	6.032E+03
	Std	5.866E+05	4.007E+06	7.450E+04	2.255E+09	5.870E+08	1.358E+09	7.225E+03	3.989E+03
	Best	1.166E+04	3.352E+03	3.425E+04	1.624E+09	3.950E+08	2.685E+08	1.941E+03	2.068E+03
	Worst	2.422E+06	2.112E+07	4.158E+05	7.161E+09	2.596E+09	4.979E+09	2.460E+04	1.975E+04
F7	Mean	2.229E+03	2.092E+03	2.178E+03	2.560E+03	2.236E+03	2.251E+03	2.119E+03	2.033E+03
	Std	8.836E+01	5.682E+01	5.066E+01	5.686E+01	3.058E+01	5.533E+01	4.945E+01	9.933E+00
	Best	2.117E+03	2.041E+03	2.098E+03	2.416E+03	2.178E+03	2.147E+03	2.056E+03	2.021E+03
	Worst	2.466E+03	2.314E+03	2.378E+03	2.614E+03	2.312E+03	2.358E+03	2.264E+03	2.055E+03
F8	Mean	2.281E+03	2.263E+03	2.281E+03	1.557E+04	2.328E+03	2.702E+03	2.229E+03	2.224E+03
	Std	5.026E+01	5.358E+01	7.720E+01	2.805E+04	5.769E+01	3.960E+02	2.161E+01	2.829E+00
	Best	2.236E+03	2.225E+03	2.232E+03	2.784E+03	2.257E+03	2.274E+03	2.222E+03	2.212E+03
	Worst	2.371E+03	2.352E+03	2.459E+03	1.486E+05	2.500E+03	4.393E+03	2.343E+03	2.228E+03
F9	Mean	2.549E+03	2.517E+03	2.501E+03	3.762E+03	2.966E+03	3.242E+03	2.481E+03	2.481E+03
	Std	3.590E+01	3.194E+01	1.246E+01	4.762E+02	1.078E+02	2.800E+02	1.924E-07	3.411E-11
	Best	2.503E+03	2.489E+03	2.484E+03	2.901E+03	2.738E+03	2.809E+03	2.481E+03	2.481E+03
	Worst	2.666E+03	2.621E+03	2.535E+03	4.705E+03	3.227E+03	4.076E+03	2.481E+03	2.481E+03
F10	Mean	4.679E+03	3.602E+03	4.004E+03	4.951E+03	3.481E+03	5.437E+03	3.710E+03	2.494E+03
	Std	1.189E+03	8.761E+02	8.479E+02	2.170E+03	8.089E+02	1.538E+03	1.084E+03	6.034E+01
	Best	2.501E+03	2.501E+03	2.501E+03	2.618E+03	2.630E+03	2.743E+03	2.500E+03	2.404E+03
	Worst	6.487E+03	5.527E+03	5.841E+03	8.669E+03	5.699E+03	6.968E+03	5.556E+03	2.641E+03
F11	Mean	3.590E+03	3.552E+03	3.039E+03	1.041E+04	8.417E+03	8.600E+03	2.993E+03	2.900E+03
	Std	7.249E+02	3.433E+02	1.707E+02	5.021E+02	4.188E+02	6.805E+02	4.445E+02	9.097E+01
	Best	3.057E+03	3.146E+03	2.663E+03	8.907E+03	7.305E+03	6.753E+03	2.600E+03	2.600E+03
	Worst	7.085E+03	4.869E+03	3.372E+03	1.070E+04	9.129E+03	9.326E+03	5.290E+03	3.000E+03
F12	Mean	3.098E+03	2.984E+03	3.167E+03	3.791E+03	3.312E+03	3.283E+03	2.943E+03	2.941E+03
	Std	9.268E+01	3.229E+01	1.601E+02	3.304E+02	8.839E+01	2.422E+02	7.513E+00	6.827E+00
	Best	2.952E+03	2.945E+03	2.983E+03	3.148E+03	3.113E+03	3.099E+03	2.933E+03	2.931E+03
	Worst	3.355E+03	3.056E+03	3.557E+03	4.733E+03	3.493E+03	4.140E+03	2.964E+03	2.965E+03
Friedman mean rank		4.61	2.95	4.18	5.85	5.20	5.60	2.20	2.11
Overall Rank		4	3	5	8	6	7	2	1

The following points explain the qualitative analysis:

- *Domain Topology:*

The first column's functions, illustrated in the two-dimensional region in Figure 8, showcase various significant typologies critical for determining the optimal performance shape for the algorithm. The algorithm can identify the most effective optimization method by analyzing the function's topology. Therefore, understanding the function's topology is crucial for developing effective optimization strategies.

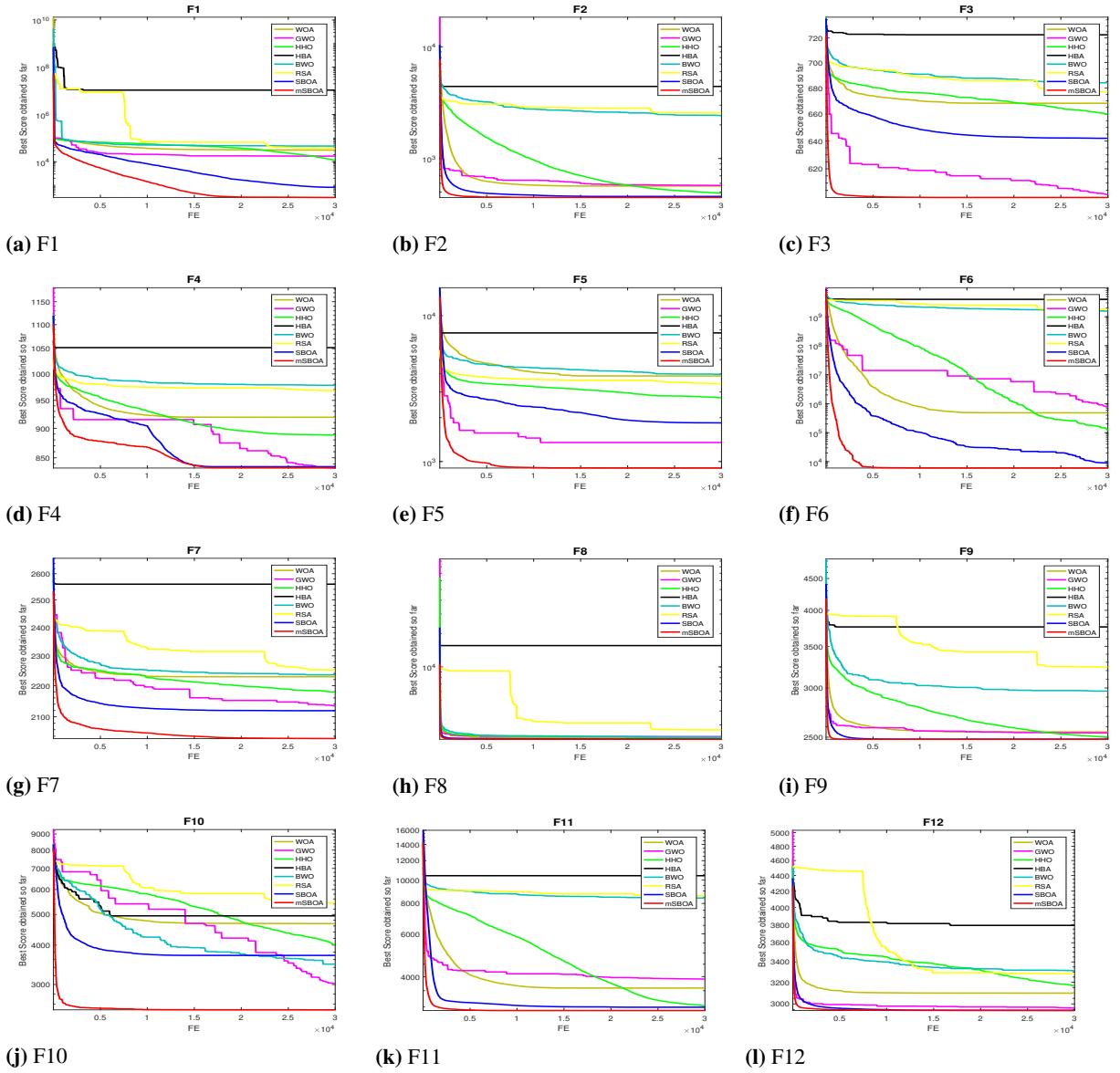
Fig. 6. The boxplot diagrams for mSBOA and its compared algorithms generated over CEC'2020 benchmarks with a dimension of 10.



- *Search History:*

The search history of the agents, from the first to the last iteration, is presented in the second column of Figure 8. Counter lines represent the search space, with a color gradient from blue to red indicating higher fitness levels. The search history demonstrates the proposed algorithm's ability to locate regions with the lowest fitness values for specific functions, indicating its effectiveness in searching the solution space. This visualization provides valuable insights into the functionality and efficiency of the proposed algorithm.

Fig. 7. Convergence plots of mSBOA and other algorithms on the CEC'2022 functions with a dimension of 10.



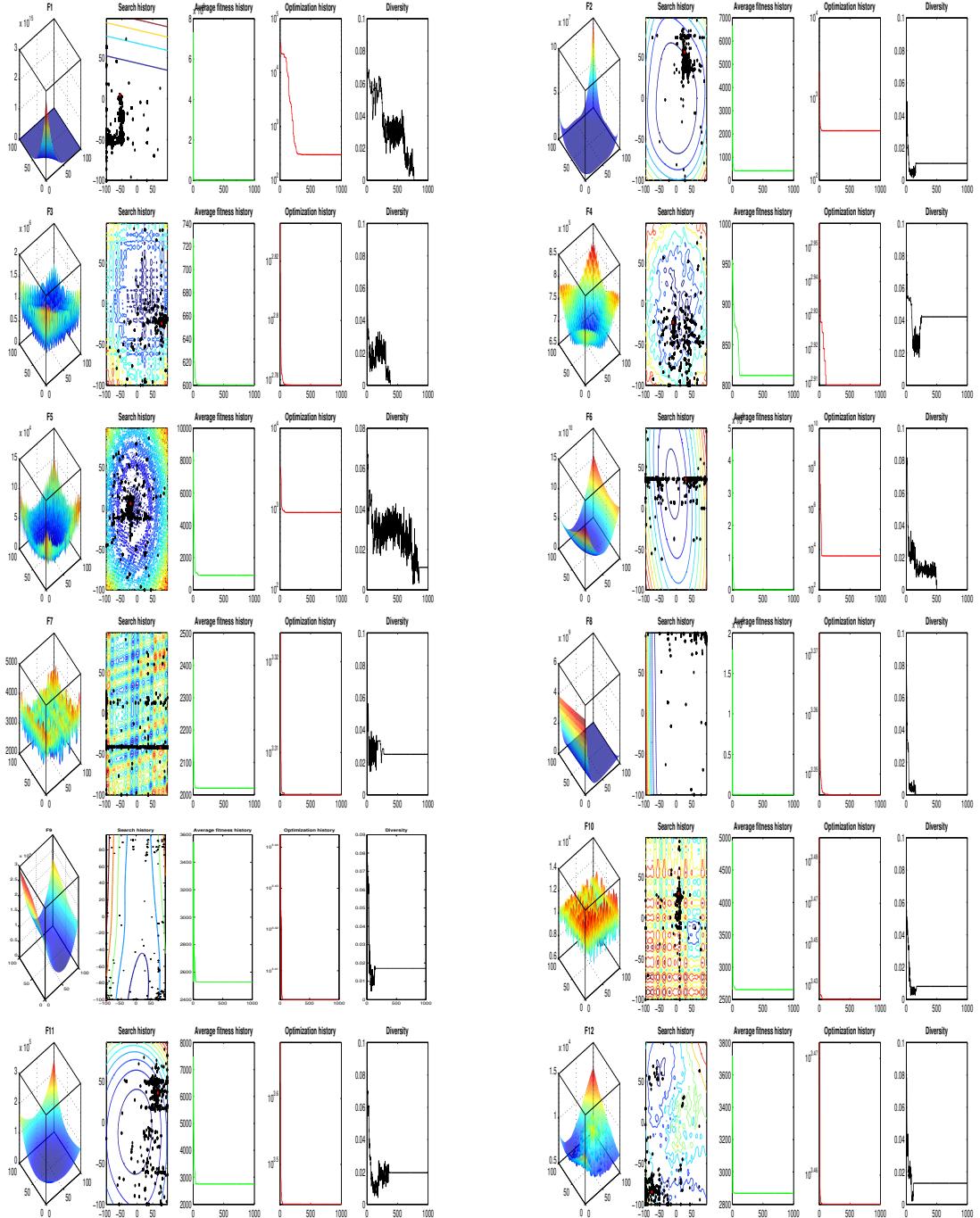
- *Comparison of Average Fitness Trajectories Across Algorithms:*

The average fitness, depicted in the third column of Figure 8, represents the average fitness value over time. This metric offers essential insights into the agents' performance and contributions to the optimization process. The graph's curves steadily decrease over time, indicating population improvement with each iteration. This enhancement supports the application of particle law updates. It demonstrates cooperative behavior among agents, showing that they collaborate well to develop better solutions as the population's fitness values decline over time.

- *Optimization History:*

The optimization improvement is shown in the fourth column of Figure 8. This figure displays the fitness outcomes over 100 iterations for each experiment, illustrating how fitness improves over time. The graph's

Fig. 8. Qualitative Metrics on CEC'2022 Test Suite



consistent downward trend suggests that the optimization method works effectively with the agents to identify the optimal solution, indicating successful collaboration among agents during the optimization process.

- **Diversity Metric:**

The last column of Figure 8 presents the diversity plot, showing the average distance traveled by the agents during the optimization process. This graph depicts the agents' exploration in their quest for the optimal solution. By calculating the average distance traveled, we can assess the diversity of solutions the agents are exploring.

6. mSBOA based multilevel image segmentation

Image segmentation is the method for classifying the different objects in the images. Images are segmented based on the selected threshold values. In this section, we will demonstrate how our suggested methods can be used for solving complex multilevel image thresholding problems. This section includes the mathematical model of the multilevel image segmentation problem along with the mathematical principle of two widely used image thresholding methods Kapur's and Otsu's methods. We have performed our study on the SCIN dataset; some sample images from the SCIN dataset are shown in Fig. 9. The multi-modal nature of the histogram makes it suitable for multilevel image segmentation problems. The outcomes achieved through the proposed methods are compared with other algorithms.

6.1. Mathematical model of image segmentation problem

The mathematical model of the image segmentation problem is :

$$\begin{aligned} R_{G_1} &\leftarrow R \text{ if } 0 \leq R < TH_1 \\ R_{G_2} &\leftarrow R \text{ if } TH_1 \leq R < TH_2 \\ R_{G_{n-1}} &\leftarrow R \text{ if } TH_{n-1} \leq R < TH_n \\ R_{G_n} &\leftarrow R \text{ if } TH_n \leq R < L - 1 \end{aligned} \quad (28)$$

where $TH_1 < TH_2 < \dots < TH_{n-1} < TH_n$ are the thresholds values which divided the image into $R_{G_1}, R_{G_2}, \dots, R_{G_n}$ regions. The optimized threshold values TH_1, TH_2, \dots, TH_n can be obtained by maximizing the following objective function

$$(TH_1^*, TH_2^*, \dots, TH_n^*) = \operatorname{argmax}(f(TH_1, TH_2, \dots, TH_n)) \quad (29)$$

In the image segmentation problem, the optimized threshold values, which are used for segmenting the image, can be achieved by optimizing an objective function $f(\cdot)$. In our work, we choose Otsu's between class variance [Otsu \(1979b\)](#), Kapur's [Kapur et al. \(1985\)](#), and entropy as the objective functions. We will formulate the objective functions in the next part of this section.

Let, I be the gray level $\{0, 1, 2, \dots, (L-1)\}$ test image with N number of pixel. If we denote the number of pixels in i^{th} gray level by f_i then $N = f_1 + f_2 + \dots + f_n$. We calculate the probability of occurrence of gray level i as follows:

$$p_i = \frac{f_i}{N}, \quad \text{where } \sum_{j=1}^L p_j = 1 \quad (30)$$

In bi-level thresholding pixels with intensity level $\{0, 1, \dots, (t-1)\}$ belong to R_0 regions and pixels $\{t, (t+1), (t+2), \dots, (L-1)\}$ belongs to R_1 .

The probabilities of these two classes are computed as

$$\omega_0 = \sum_{i=0}^{t-1} p_i \quad \text{and} \quad \omega_1 = \sum_{i=t}^{L-1} p_i \quad (31)$$

respectively. For multilevel thresholding where images are separated into multiple regions, the probability of each region is defined as

$$\omega_0 = \sum_{i=0}^{t_1-1} p_i, \omega_1 = \sum_{i=t_1}^{t_2-1} p_i, \omega_2 = \sum_{i=t_2}^{t_3-1} p_i, \dots, \text{and} \quad \omega_n = \sum_{i=t_n}^{L-1} p_i \quad (32)$$

The new set of probability distribution for bi-level distribution can be obtained by

$$p_{R_1} = \frac{p_0}{\omega_0}, \frac{p_1}{\omega_0}, \dots, \frac{p_{t-1}}{\omega_0} \quad \text{and} \quad p_{R_2} = \frac{p_t}{\omega_1}, \frac{p_{t+1}}{\omega_1}, \dots, \frac{p_{(L-1)}}{\omega_1} \quad (33)$$

In the case of multilevel following are the probability distribution of different regions

$$\begin{aligned} p_{R_1} &= \frac{p_0}{\omega_0}, \frac{p_1}{\omega_0}, \dots, \frac{p_{t_1-1}}{\omega_0}, p_{R_2} = \frac{p_t}{\omega_1}, \frac{p_{t+1}}{\omega_1}, \dots, \frac{p_{t_2-1}}{\omega_1}, \\ p_{R_3} &= \frac{p_{t_2}}{\omega_2}, \frac{p_{t_2+1}}{\omega_2}, \dots, \frac{p_{t_2-1}}{\omega_2}, p_{R_n} = \frac{p_{t_3}}{\omega_n}, \frac{p_{t_3+1}}{\omega_n}, \dots, \frac{p_{(L-1)}}{\omega_{n-1}} \end{aligned} \quad (34)$$

6.2. Kapur's objective function

Kapur's objective function (in terms of entropy) [Kapur et al. \(1985\)](#) for dividing an image into $n + 1$ different regions using n number thresholds is defined as follows:

$$f(t) = H_0 + H_1 + H_2 + \dots + H_n \quad (35)$$

where $t = [t_1, t_2, t_3, \dots, t_n]$

$$\left\{ \begin{array}{l} H_0 = - \sum_{i=0}^{t_1-1} \frac{p_i}{\omega_0} \ln \frac{p_i}{\omega_0} \\ H_1 = - \sum_{i=t_1}^{t_2-1} \frac{p_i}{\omega_1} \ln \frac{p_i}{\omega_1} \\ H_2 = - \sum_{i=t_2}^{t_3-1} \frac{p_i}{\omega_2} \ln \frac{p_i}{\omega_2} \\ \vdots \\ H_n = - \sum_{i=t_{n-1}}^{L-1} \frac{p_i}{\omega_{n-1}} \ln \frac{p_i}{\omega_{n-1}} \end{array} \right. \quad (36)$$

H_0, H_1, \dots, H_n , are the Kapur's entropy for segmented region. The set of optimum thresholds(t^*) using Kapur's entropy is obtained by maximizing the following function:

$$t^* = [t_1^*, t_2^*, t_3^*, \dots, t_n^*] = \text{argmax}(f(t)), \quad (37)$$

6.3. Otsu's objective function

The Otsu's objective function is given by,

$$f(t) = \sum_{i=0}^n \sigma_i \quad (38)$$

$$\begin{aligned} \sigma_0 &= \omega_0(\mu_0 - \mu_T)^2, \sigma_1 = \omega_1(\mu_1 - \mu_T)^2 \\ \sigma_j &= \omega_j(\mu_j - \mu_T)^2, \sigma_n = \omega_n(\mu_n - \mu_T)^2 \end{aligned} \quad (39)$$

where μ_T is the mean intensity of the whole image. The cumulative probability of each region can be calculated as:

$$\mu_0 = \sum_{i=0}^{t_1-1} \frac{ip_i}{\omega_i}, \mu_1 = \sum_{i=t_1}^{t_2-1} \frac{ip_i}{\omega_i}, \mu_j = \sum_{i=t_j}^{t_{j+1}-1} \frac{ip_i}{\omega_i}, \mu_n = \sum_{i=t_n}^{L-1} \frac{ip_i}{\omega_i} \quad (40)$$

The set of optimal thresholds(t^*) can be obtained by maximizing the objective function of the Eqn. (38)

$$t^* = [t_1^*, t_2^*, t_3^*, \dots, t_n^*] = \text{argmax}(f(t)) \quad (41)$$

6.4. Quantitative and qualitative analysis

In multilevel thresholding calculation, each m – m-level thresholding is considered as an independent process. The proposed algorithms are applied separately for each image to find 5, 6, and 7 levels of thresholding considering the Otsu and Kapur method of segmentation technique as the objective function. In the RGB color space of color images, the optimum threshold value is calculated separately for each red, green, and blue band. The optimum threshold values together with the maximum objective function values of the red, green, and blue bands of different segmentation methods are recorded. Table 7 and Table 8 show the optimal threshold value of selected images for the red and green band using Kapur and Otsu as the objective function. From the above table, it is established that the projected mSBOA outflanks the other algorithms in terms of the maximum objective function value.

Segmented images of the proposed algorithm along with the other algorithms are shown in Figs. 10 and 11. Images are reconstructed using the optimized red, green, the blue bands of the RGB color space. The red band is optimized using the Kapur segmentation method and the green and the blue band are optimized by Otsu's method of segmentation.

Peak signal-to-noise ratio (PSNR), structural similarity index metric (SSIM) Wang et al. (2004) and feature similarity index metric (FSIM) Zhang et al. (2011) are the three quality matrices utilized in this study to measure the quality of the segmented images. FSIM is used to find feature similarity of two images $f_1(X)$ and $f_2(X)$ and is defined by,

$$FSIM = \frac{\sum_{X_\epsilon \Omega} S_L(X) PC_m(X)}{\sum_{X_\epsilon \Omega} PC_m(X)} \quad (42)$$

where Ω represents the spacial domain of entire image and $S_L(X) = S_{pC}(X)S_G(X)$. $S_{pC}(X)$ and $S_G(X)$ are given by,

$$\begin{aligned} S_{pC}(X) &= \frac{2PC_1(X)PC_2(X) + T_1}{PC_1^2(X) + PC_2^2 + T_1} \\ S_G(X) &= \frac{2G_1(X)G_2(X) + T_2}{G_1^2(X) + G_2^2(X) + T_2} \end{aligned} \quad (43)$$

where PC_1 and PC_2 are the phase congruence maps taken out from two images $f_1(X)$, and $f_2(X)$, respectively; T_1 and T_2 are constants. The values of $T_1 = 0.85$, and $T_2 = 160$. Higher value of FSIM indicates better segmentation.

Calculated PSNR, SSIM, and FSIM esteem of the segmented images in reference to the original images are recorded and appear in Table 9. The result demonstrates that the segmented images' quality is improved after integrating oppositional with the EHO instead of separately integrating the chaotic and oppositional with the EHO and SOS.

Calculated PSNR, SSIM, and FSIM esteem of the segmented images in reference to the original images are recorded and appear in Table 4. The result demonstrates that the quality of the segmented images is improved after integrating both Orthogonal and Opposition learning with the SBOA.

Fig. 9. Sample SCIN dataset with their histograms for RGB brands

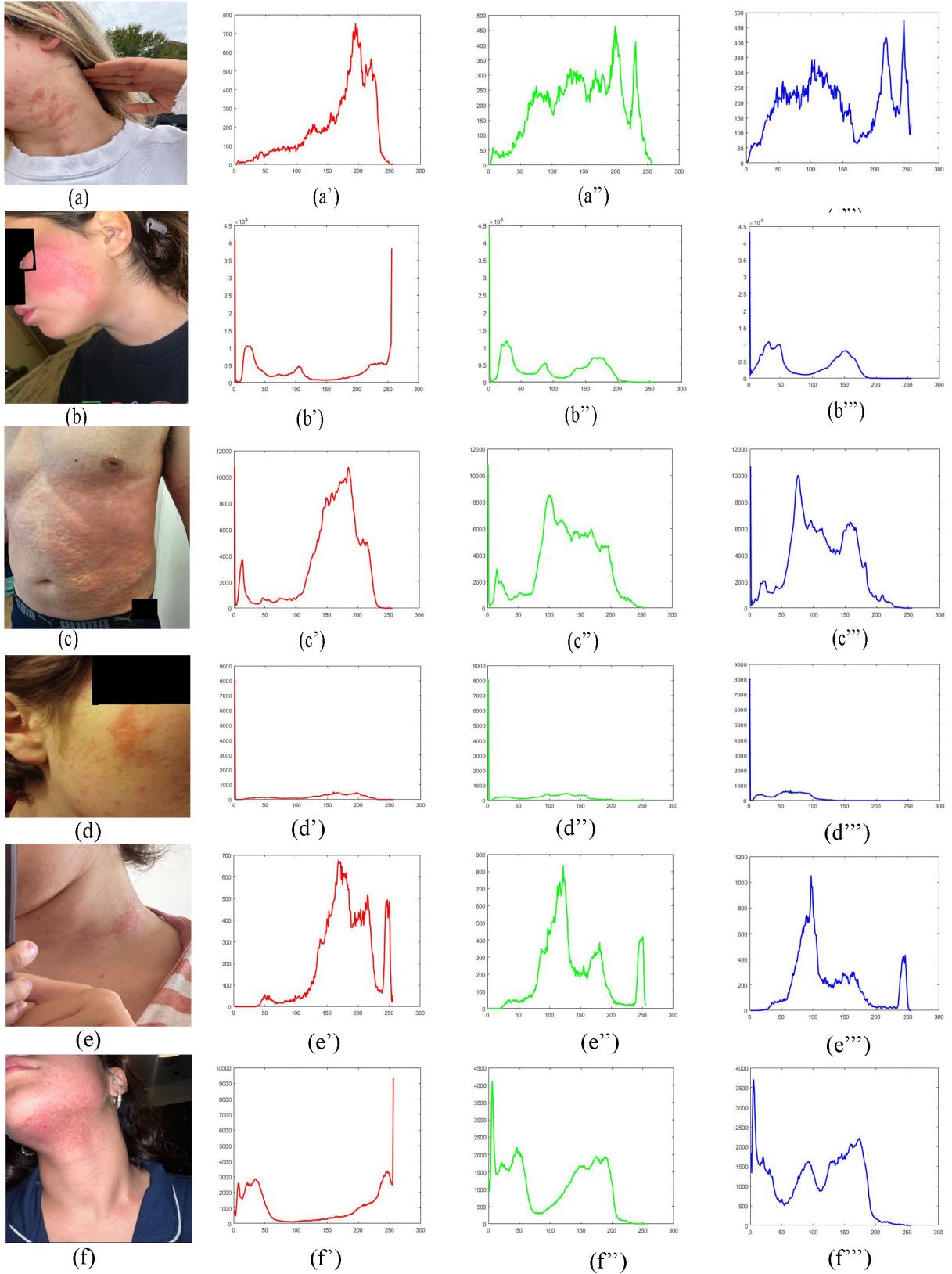


Table 7 Fitness values of Kapur's entropy along with the corresponding threshold values for red band for different algorithms

Image	K	mSBOA	SBOA	WOA	GWO	BWO	HHO	mSBOA	SBOA	WOA	GWO	BWO	HHO
SCIN1	5	21.1737	21.1731	21.1731	21.1731	21.1731	21.1731	46 93 133 178 212	46 92 133 178 212	46 92 133 178 212	46 92 133 178 212	46 92 133 178 212	46 92 133 178 212
	6	23.7985	23.7964	23.7947	23.7958	23.7958	23.7964	38 67 100 133 178	41 67 101 134 178	38 67 101 134 178	42 70 101 134 178	42 70 101 134 178	41 67 101 134 178
	7	26.2010	26.1853	26.1865	26.1843	26.184	26.1853	38 67 100 133 174	37 65 99 134 174 198	37 64 97 133 174 198	37 65 99 133 174 197	38 67 98 133 174 197	37 65 99 134 174 198
SCIN2	5	21.5771	21.5750	21.5733	21.5748	21.576	21.575	48 89 129 168 207	46 86 127 168 207	46 86 127 168 208	49 88 128 167 206	49 90 130 168 206	46 86 127 168 207
	6	24.1855	24.1847	24.1833	24.1847	24.1847	24.1838	41 74 108 142 177	41 75 108 143 177	41 75 109 144 180	41 75 108 143 177	41 75 108 143 177	41 74 110 144 178
	7	26.6605	26.6592	26.6596	26.6581	26.6592	26.6581	37 67 97 127 157 186	37 67 98 129 159 189	38 68 98 128 158 163	37 67 97 131 159 189	37 67 98 129 159 189	37 67 97 131 159 189
SCIN3	5	20.0889	20.0875	20.0875	20.0877	20.0875	20.0877	44 93 140 182 233	44 93 140 182 233	46 94 140 183 233	44 93 140 183 233	46 94 140 183 233	44 93 140 183 233
	6	22.8738	22.8735	22.8722	22.8735	22.8722	22.8735	42 83 126 167 202	42 82 125 167 202	42 82 125 166 201	42 82 125 167 202	42 82 125 166 201	41 82 125 167 202
	7	25.5330	25.5270	25.5246	25.5258	25.5262	25.5270	36 67 103 138 169	36 67 102 138 170	35 68 104 137 170	36 68 102 139 170	35 66 100 136 168	36 67 102 138 170
SCIN4	5	20.1768	20.1768	20.1768	20.1768	20.1768	20.1768	108 139 169 199 227	108 139 169 199 227	108 139 169 199 227	108 139 169 199 227	108 139 169 199 227	108 139 169 199 227
	6	22.5014	22.5006	22.5006	22.5001	22.5005	22.5006	83 109 138 169 198	83 110 141 169 200	83 110 141 169 200	83 111 141 171 199	83 109 141 170 199	83 110 141 169 200
	7	24.7415	24.7408	24.7408	24.7394	24.7379	24.7408	83 108 134 159 181	83 107 130 153 178	83 107 130 153 178	82 106 128 150 174	81 107 132 154 178	83 107 130 153 178
SCIN5	5	20.8670	20.8654	20.8654	20.8654	20.8654	20.8654	57 90 137 181 214	57 90 137 181 214	58 93 138 181 214	58 93 138 181 214	58 93 138 181 214	58 93 138 181 214
	6	23.4229	23.4219	23.4219	23.4189	23.4219	23.4165	57 90 133 162 192	57 90 134 162 192	57 90 134 162 192	57 90 134 164 192	57 90 134 162 192	58 90 133 165 192
	7	25.8582	25.8464	25.8508	25.8464	25.8405	25.8464	55 86 110 136 164	53 84 109 135 166	56 86 112 136 166	53 84 109 135 166	58 87 112 137 167	53 84 109 135 166
SCIN6	5	20.9682	20.9682	20.9676	20.9676	20.9676	20.9682	44 75 104 143 180	44 75 104 143 180	44 75 104 143 181	44 75 104 143 181	44 75 104 143 180	44 75 104 143 180
	6	23.4602	23.4596	23.4588	23.4596	23.4588	23.4596	43 72 101 131 159	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159
	7	25.7929	25.7835	25.7835	25.7808	25.7808	25.7835	27 54 81 104 131 161	27 55 81 106 135 160	27 55 81 106 135 160	28 55 83 107 135 160	28 55 83 107 135 160	27 55 81 106 135 160
SCIN7	5	20.9682	20.9682	20.9676	20.9676	20.9676	20.9682	44 75 104 143 180	44 75 104 143 180	44 75 104 143 181	44 75 104 143 181	44 75 104 143 180	44 75 104 143 180
	6	23.4602	23.4596	23.4588	23.4596	23.4588	23.4596	43 72 101 131 159	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159
	7	25.7929	25.7835	25.7835	25.7808	25.7808	25.7835	27 54 81 104 131 161	27 55 81 106 135 160	27 55 81 106 135 160	28 55 83 107 135 160	28 55 83 107 135 160	27 55 81 106 135 160
SCIN8	5	20.9682	20.9682	20.9676	20.9676	20.9676	20.9682	44 75 104 143 180	44 75 104 143 180	44 75 104 143 181	44 75 104 143 181	44 75 104 143 180	44 75 104 143 180
	6	23.4602	23.4596	23.4588	23.4596	23.4588	23.4596	43 72 101 131 159	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159
	7	25.7929	25.7835	25.7835	25.7808	25.7808	25.7835	27 54 81 104 131 161	27 55 81 106 135 160	27 55 81 106 135 160	28 55 83 107 135 160	28 55 83 107 135 160	27 55 81 106 135 160
SCIN9	5	20.9682	20.9682	20.9676	20.9676	20.9676	20.9682	44 75 104 143 180	44 75 104 143 180	44 75 104 143 181	44 75 104 143 181	44 75 104 143 180	44 75 104 143 180
	6	23.4602	23.4596	23.4588	23.4596	23.4588	23.4596	43 72 101 131 159	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159
	7	25.7929	25.7835	25.7835	25.7808	25.7808	25.7835	27 54 81 104 131 161	27 55 81 106 135 160	27 55 81 106 135 160	28 55 83 107 135 160	28 55 83 107 135 160	27 55 81 106 135 160
SCIN10	5	20.9682	20.9682	20.9676	20.9676	20.9676	20.9682	44 75 104 143 180	44 75 104 143 180	44 75 104 143 181	44 75 104 143 181	44 75 104 143 180	44 75 104 143 180
	6	23.4602	23.4596	23.4588	23.4596	23.4588	23.4596	43 72 101 131 159	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159	41 71 100 129 158	43 73 101 130 159
	7	25.7929	25.7835	25.7835	25.7808	25.7808	25.7835	27 54 81 104 131 161	27 55 81 106 135 160	27 55 81 106 135 160	28 55 83 107 135 160	28 55 83 107 135 160	27 55 81 106 135 160

Table 8 Fitness values of Otsu's method along with the corresponding threshold values of the green band for different algorithms

Image	K	mSBOA	SBOA	WOA	GWO	BWO	HHO	mSBOA	SBOA	WOA	GWO	BWO	HHO			
SCIN1	5	3165.9737 3165.7688 3165.7551 3165.7551 3165.7551 3165.7551	61 102 141 178 212	59 102 141 176 213	61 103 141 178 214	61 103 141 178 214	61 103 141 178 214	61 103 141 178 214	61 103 141 178 214	53 90 122 153 184	54 90 122 153 184	54 91 123 154 184	54 91 123 154 184	54 91 123 154 184	54 91 123 154 184	
	6	3201.0126 3201.0101 3201.007 3201.0068 3201.0068 3201.0068	215	214	215	215	215	215	47 77 104 131 159	47 79 105 131 159	47 78 106 132 159	49 80 107 132 159	47 78 106 132 159	49 80 107 132 159		
	7	3221.0771 3221.0349 3221.0228 3221.0149 3221.0228 3221.0149	187 216	186 216	185 215	186 215	185 215	186 215					186 215	186 215		
SCIN2	5	1871.1365 1871.1135 1871.0859 1871.0859 1871.0859 1871.0859	48 87 128 157 180	46 88 128 157 180	48 88 128 156 180	48 88 128 156 180	48 88 128 156 180	48 88 128 156 180	48 88 128 156 180	48 87 124 149 168	49 87 124 149 168	49 87 124 149 168	48 86 124 149 168	48 88 128 156 180	48 86 124 149 168	
	6	1886.8911 1886.874 1886.8715 1886.8715 1886.8655 1886.8655	185	185	185	185	185	185	49 86 120 142 159	48 85 121 143 159	48 87 122 144 160	49 87 122 143 160	49 87 122 143 160	49 87 122 143 160	185	185
	7	1895.3844 1895.3621 1895.3385 1895.3352 1895.3352 1895.3352	174 188	174 188	175 188	175 188	175 188	175 188					175 188	175 188		
SCIN3	5	3092.4838 3092.4358 3092.4326 3092.4326 3092.4326 3092.4326	18 54 88 116 146	18 53 86 115 147	17 52 87 115 146	17 52 87 115 146	17 52 87 115 146	17 52 87 115 146	17 52 87 115 146	18 52 84 111 137 165	18 51 84 111 137 165	18 52 85 112 138 165	17 51 84 111 137 165	17 51 84 111 137 165	17 52 87 115 146	
	6	3119.7953 3119.7837 3119.7831 3119.7807 3119.7807 3119.7807	174 89 79 103 122 142	174 78 102 123 143	17 48 77 101 120 141	17 50 80 103 122 142	17 50 80 103 122 142	16 47 76 100 121 141	16 47 76 100 121 141	168	168	166	166	166	166	
	7	3133.0705 3133.0061 3132.9954 3132.9662 3132.934 3132.934	168	168	166	166	166	166					166	166		
SCIN4	5	2358.1386 2358.0652 2358.0102 2358.0102 2358.0102 2358.0102	73 106 134 165 212	73 107 135 166 211	74 107 134 165 214	74 107 134 165 214	74 107 134 165 214	74 107 134 165 214	74 107 134 165 214	68 99 118 142 170	68 98 118 142 170	68 98 118 142 170	67 98 117 141 170	67 98 117 141 170	74 107 134 165 214	
	6	2378.4183 2378.4151 2378.4029 2378.382 2378.382 2378.382	214	214	215	214	214	214	66 95 114 135 160	67 95 114 135 159	66 95 115 135 158	64 95 114 134 158	64 95 114 134 158	64 95 114 134 158	214	214
	7	2392.3674 2392.3326 2392.3269 2392.3176 2392.3176 2392.3176	181 217	183 220	182 218	182 217	182 217	179 217					179 217	179 217		
SCIN5	5	991.2277 991.2072 991.2072 991.2072 991.2072 991.2072	22 52 75 91 106	22 51 75 91 106	22 51 75 91 106	22 51 75 91 106	22 51 75 91 106	22 51 75 91 106	42 99 141 169 193	41 97 141 169 194	42 97 142 170 194	41 99 142 169 194	41 99 142 169 194	41 99 142 169 194	22 51 75 91 106	
	6	4228.0061 4227.9997 4227.9553 4227.9356 4227.9356 4227.9356	220	221	221	220	220	200 224	30 75 119 152 177	33 78 118 150 177	33 78 118 153 177	31 73 116 150 177	31 73 116 150 177	31 73 116 150 177	220	220
	7	4248.3509 4248.2704 4248.1844 4248.1662 4248.1662 4248.1662	200 224	200 225	199 224	199 224	199 224	200 224					200 224	200 224		
SCIN6	5	4047.2006 4047.1312 4047.1056 4047.1056 4047.1056 4047.1056	27 61 108 147 173	27 61 108 147 172	26 59 106 146 173	26 59 106 146 173	26 59 106 146 173	26 59 106 146 173	17 42 72 111 147 173	17 42 72 111 148 174	16 42 72 111 148 173	16 41 71 110 148 173	16 41 71 110 148 173	16 41 71 110 148 173	26 59 106 146 173	
	6	4071.3296 4071.3274 4071.3087 4071.2788 4071.2788 4071.2788	164 1 69 99 127 153	164 60 99 100 128 153	164 70 101 130 154	164 70 101 130 154	164 70 101 130 154	175	175	176	176	175	175	175	175	
	7	4083.9527 4083.8873 4083.8732 4083.8584 4083.8584 4083.8584	175	177	176	176	176	175					175	175		
SCIN7	5	2371.8056 2371.7608 2371.7492 2371.7492 2371.7492 2371.7492	51 93 121 151 183	50 93 120 151 183	47 91 119 149 181	47 91 119 149 181	47 91 119 149 181	47 91 119 149 181	40 79 109 135 161	38 78 108 134 161	39 78 108 133 160	39 80 109 135 162	39 80 109 135 162	39 80 109 135 162	47 91 119 149 181	
	6	2399.2302 2399.2167 2399.2084 2399.1935 2399.1915 2399.1915	188	189	189	189	189	188					189	189		
	7	2417.0115 2416.9548 2416.9054 2416.8372 2416.8372 2416.8372	39 77 106 129 152	38 78 107 131 154	39 78 105 128 151	38 76 105 128 152	38 76 105 128 152	38 76 105 128 152	177 203	178 203	176 203	175 203	175 203	175 203	38 76 105 128 152	
SCIN8	5	1690.1674 1690.0913 1690.0752 1690.0752 1690.0752 1690.0752	33 74 103 127 152	33 75 103 128 151	34 74 102 126 151	34 74 102 126 151	34 74 102 126 151	34 74 102 126 151	33 74 101 122 140	33 74 101 122 140	32 75 82 107 129 153	34 74 102 122 140	34 74 102 122 140	34 74 102 126 151	34 74 102 126 151	
	6	1704.9455 1704.9093 1704.8947 1704.8882 1704.8882 1704.8882	159	161	160	160	160	160					160	160		
	7	1718.6067 1718.5674 1718.4926 1718.4926 1718.4926 1718.4926	26 55 80 104 123 141	27 56 80 104 124 141	28 56 81 104 123 140	28 56 81 104 123 140	28 56 81 104 123 140	28 56 81 104 123 140	161	160	160	160	160	160	28 56 81 104 123 140	
SCIN9	5	4427.9331 4427.9245 4427.9245 4427.9177 4427.9177 4427.9177	32 74 118 149 178	32 75 118 151 177	32 75 118 151 178	32 75 117 151 178	32 75 117 151 178	32 75 117 151 178	23 47 82 123 153 180	23 47 82 123 153 179	23 47 83 124 153 179	23 47 83 124 153 179	23 47 83 124 153 179	32 75 117 151 178	32 75 117 151 178	
	6	4450.4588 4450.4536 4450.4149 4450.4078 4450.4078 4450.4078	21 43 74 110 138 161	23 45 75 111 137 160	22 44 73 110 137 162	22 45 77 111 138 161	22 45 77 111 138 161	22 45 77 111 138 161	184	183	184	185	185	185	23 47 83 124 153 179	
	7	4465.9767 4465.9378 4465.8438 4465.8425 4465.8425 4465.8425	203	204	203	205	205	205					205	205		
SCIN10	5	3393.9603 3393.8851 3393.8491 3393.8122 3393.8122 3393.8122	40 75 112 156 200	39 75 111 154 198	39 76 114 158 199	38 73 110 156 200	38 73 110 156 200	38 73 110 156 200	35 65 93 126 163 202	35 65 92 125 162 201	35 65 92 125 163 201	35 65 92 125 163 201	35 65 92 125 163 201	38 73 110 156 200	38 73 110 156 200	
	6	3429.787 3429.7789 3429.741 3429.7382 3429.7382 3429.7382	34 62 86 113 142 171	34 62 86 114 142 171	34 62 86 114 142 170	34 63 86 112 142 173	34 63 86 112 142 173	34 63 86 112 142 173	203	204	203	205	205	205	34 63 86 112 142 173	
	7	3449.7557 3449.7106 3449.6123 3449.5764 3449.5764 3449.5764	203	204	203	205	205	205					205	205		

Table 9 The PSNR, FSIM, and SSIM values of images segmented by Kapur method.

		PSNR (dB)						SSIM						FSIM					
Image	K	mSBOA	SBOA	WOA	GWO	BWO	HHO	mSBOA	SBOA	WOA	GWO	BWO	HHO	mSBOA	SBOA	WOA	GWO	BWO	HHO
SCIN1	5	30.8733	30.8502	30.8502	30.8502	30.8502	30.8147	0.9327	0.921	0.921	0.921	0.921	0.9323	0.9991	0.9984	0.9984	0.9984	0.9984	0.9985
	6	25.7529	24.6639	24.6628	24.6639	24.6628	25.2635	0.9093	0.8924	0.8924	0.8924	0.8924	0.9026	0.9935	0.9922	0.9921	0.9922	0.9921	0.993
	7	26.1434	25.9067	25.7688	25.9067	25.7688	25.7076	0.9125	0.9029	0.9101	0.9029	0.9101	0.9064	0.9938	0.9926	0.9934	0.9926	0.9934	0.9926
SCIN2	5	39.9653	39.6708	39.6708	39.6708	39.6708	38.8753	0.9922	0.9916	0.9916	0.9916	0.9916	0.9788	0.9972	0.998	0.998	0.998	0.998	0.9886
	6	39.5876	39.2585	39.1411	39.1411	39.2585	38.7345	0.9916	0.9915	0.9906	0.9906	0.9915	0.975	0.9983	0.9987	0.9983	0.9983	0.9987	0.9973
	7	39.6323	39.4927	39.3563	39.4927	39.3563	39.2078	0.9909	0.9908	0.9901	0.9908	0.9901	0.9902	0.9983	0.9985	0.9982	0.9985	0.9982	0.9982
SCIN3	5	25.2389	25.083	25.083	25.083	25.083	25.0815	0.9532	0.9528	0.9528	0.9528	0.9528	0.9528	0.9935	0.9931	0.9931	0.9931	0.9931	0.9931
	6	21.9429	21.6819	21.6819	21.6819	21.6819	21.4422	0.9319	0.9277	0.9277	0.9277	0.9277	0.9263	0.9888	0.9886	0.9886	0.9886	0.9886	0.9885
	7	27.8833	27.6272	27.6272	27.6272	27.6272	27.5698	0.9720	0.9691	0.9691	0.9691	0.9691	0.969	0.9961	0.995	0.995	0.995	0.995	0.9949
SCIN4	5	28.9325	28.9145	28.9145	28.9145	28.9145	28.8685	0.8703	0.8633	0.8633	0.8633	0.8633	0.8696	0.9983	0.9972	0.9972	0.9972	0.9972	0.997
	6	23.9652	23.9095	23.8446	23.8342	23.8446	23.7439	0.8777	0.8763	0.8742	0.8751	0.8742	0.8735	0.988	0.9878	0.9875	0.9874	0.9875	0.9867
	7	24.9501	24.9484	24.9421	24.9421	24.9421	24.1803	0.8967	0.8955	0.8952	0.8952	0.8952	0.8815	0.9901	0.9899	0.9898	0.9898	0.9898	0.9897
SCIN5	5	24.5732	23.6459	23.6459	23.6459	23.6459	23.6211	0.9297	0.9198	0.9198	0.9198	0.9198	0.9168	0.989	0.9813	0.9813	0.9813	0.9813	0.9766
	6	25.3038	25.0326	25.0326	25.0264	25.0326	25.0091	0.9368	0.9355	0.9355	0.9345	0.9355	0.9353	0.9864	0.9836	0.9836	0.9831	0.9836	0.9835
	7	25.2355	25.1934	25.1642	25.1642	25.1642	24.6427	0.9427	0.9421	0.9413	0.9413	0.9413	0.9159	0.9829	0.9828	0.9829	0.9829	0.9829	0.9831
SCIN6	5	31.1045	30.9871	30.9871	30.9871	30.9871	30.9871	0.9025	0.9023	0.9023	0.9023	0.9023	0.9018	0.9979	0.9965	0.9965	0.9965	0.9965	0.9974
	6	26.2567	26.3022	26.2567	26.2567	26.0869	26.03527	0.8892	0.8882	0.8892	0.8892	0.8833	0.872	0.9900	0.9901	0.99	0.99	0.99	0.9883
	7	24.3895	23.5971	3.5864	23.5971	3.5864	23.4986	0.9191	0.8689	0.8651	0.8689	0.8651	0.8665	0.9953	0.9545	0.9543	0.9545	0.9543	0.952
SCIN7	5	28.8837	28.8702	28.8702	28.8702	28.8702	28.8586	0.8661	0.8658	0.8658	0.8658	0.8658	0.8641	0.9969	0.9965	0.9965	0.9965	0.9965	0.9964
	6	24.3449	24.3186	24.2354	24.1376	24.3186	23.5698	0.8676	0.8683	0.8655	0.8627	0.8683	0.8534	0.9862	0.9864	0.986	0.9855	0.9864	0.983
	7	25.2467	25.0112	25.1436	25.0112	25.1436	25.0023	0.8890	0.8825	0.8856	0.8825	0.8856	0.8821	0.9884	0.9869	0.9873	0.9869	0.9873	0.9853
SCIN8	5	24.9341	24.731	24.731	24.731	24.731	24.6869	0.9328	0.9312	0.9312	0.9312	0.9312	0.9297	0.9663	0.9673	0.9673	0.9673	0.9673	0.9657
	6	25.9674	25.7472	5.8271	25.7869	25.7869	25.1559	0.9455	0.9438	0.9447	0.9442	0.9442	0.9368	0.9723	0.9713	0.9718	0.9715	0.9715	0.969
	7	26.7191	26.6888	26.6184	26.6888	26.6888	26.5734	0.9578	0.9559	0.9554	0.9559	0.9559	0.9555	0.9777	0.975	0.9749	0.975	0.975	0.9746
SCIN9	5	25.4334	25.2748	25.2748	25.2748	25.2748	25.2228	0.8344	0.8311	0.8311	0.8311	0.8311	0.8309	0.9856	0.9845	0.9845	0.9845	0.9845	0.9846
	6	26.2991	26.2982	26.2982	26.2982	26.2982	26.1999	0.8497	0.8495	0.8495	0.8483	0.8495	0.8481	0.9872	0.9875	0.9875	0.9875	0.9875	0.9869
	7	27.3078	27.2684	27.1983	27.1983	27.1983	27.1967	0.8634	0.8634	0.8634	0.8634	0.8634	0.8611	0.9894	0.9891	0.9892	0.9892	0.9892	0.9889
SCIN10	5	24.1856	24.1856	24.1856	24.1856	24.1856	24.1856	0.9121	0.9121	0.9121	0.9121	0.9121	0.9121	0.9906	0.9906	0.9906	0.9906	0.9906	0.9906
	6	26.2058	26.1286	26.1286	26.1286	26.1286	25.3284	0.9403	0.9407	0.9407	0.9407	0.9407	0.9326	0.9936	0.9936	0.9936	0.9936	0.9936	0.9921
	7	27.5295	27.0344	27.0344	27.0344	27.0344	27.0118	0.9513	0.944	0.944	0.944	0.944	0.9401	0.9952	0.9949	0.9949	0.9949	0.9949	0.9942

6.5. Stability of the algorithm

Due to the stochastic nature of the meta-heuristic search, the optimal result through a single run may not be achieved, therefore a similar experiment is repeated to calculate the optimal result as well as the mean and standard deviation of the algorithm to verify the stability of the algorithms.

The higher the mean value and lower the SD value, is the stability factor of any heuristic algorithm. Thus, the proposed mSBOA and other algorithms are rehashed 30 times for each algorithm of each m-level, and the recorded mean and SD value are shown in Table 10. The result suggests the stability of the proposed algorithm over the other algorithms.

Fig. 10. The diagrams for mSBOA and its compared algorithms .

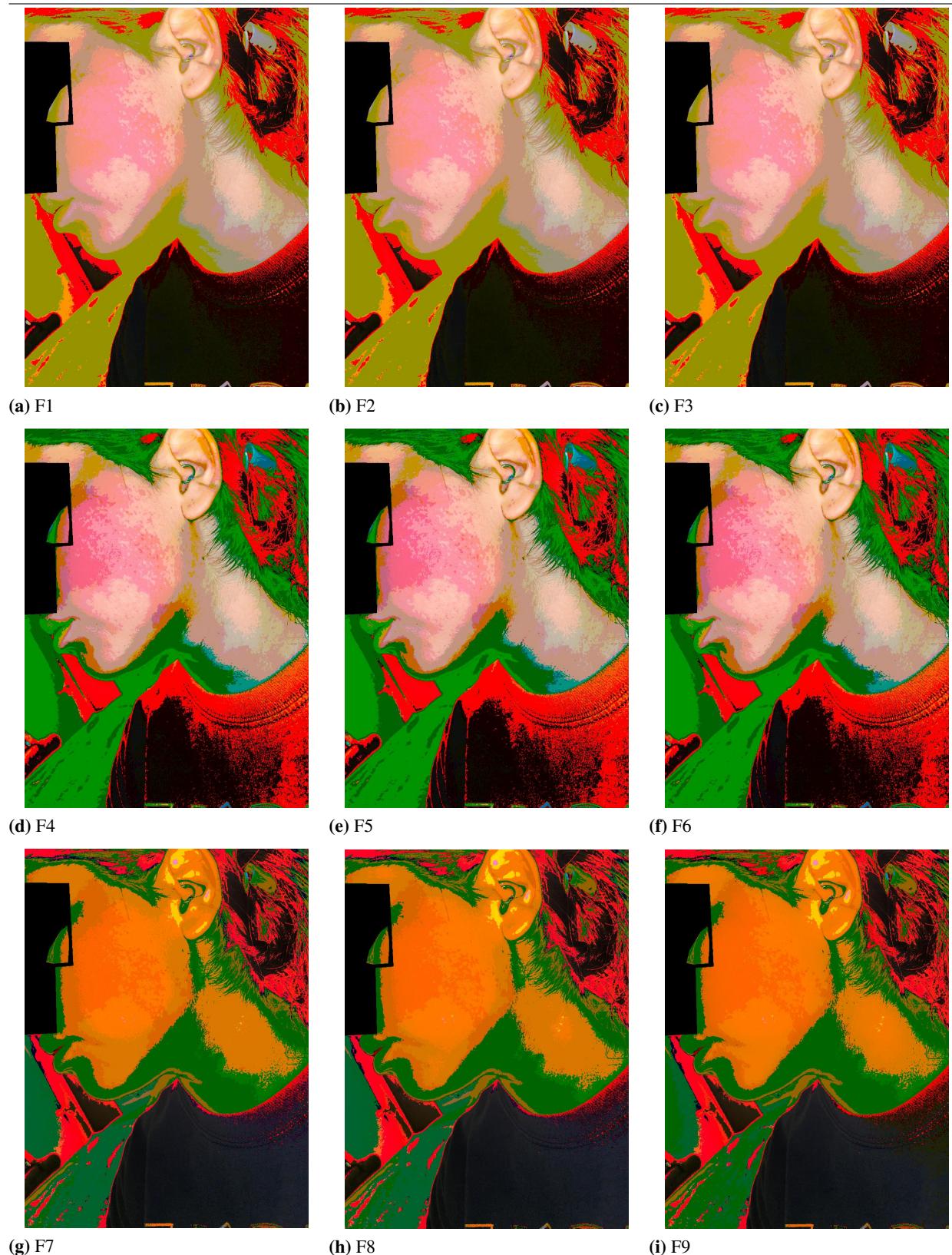


Fig. 11. The diagrams for mSBOA and its compared algorithms .

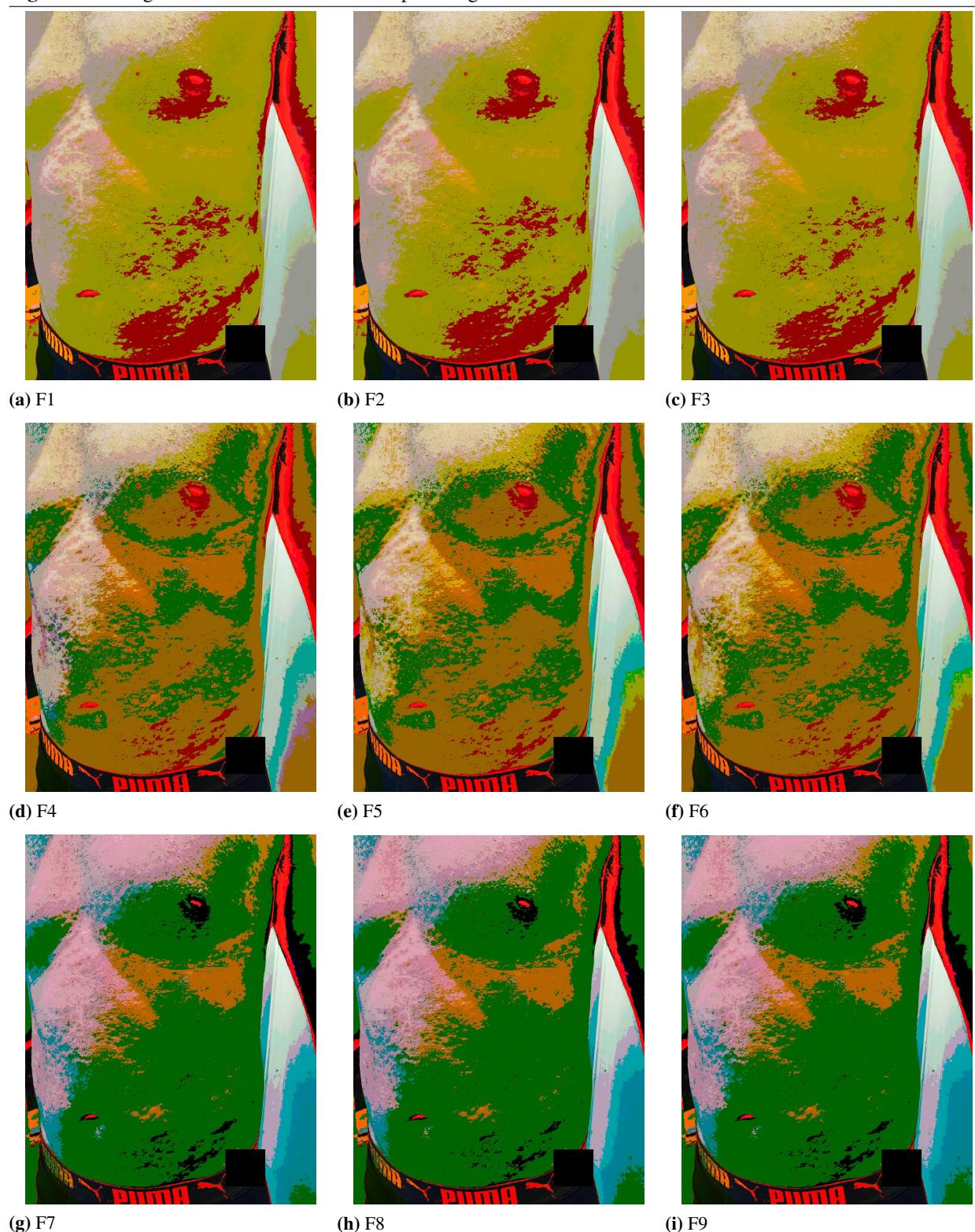


Table 10 Mean and SD values of Otsu's method along with the corresponding threshold values green red band for different algorithms

Image	K	mSBOA	SBOA	WOA	GWO	BWO	HHO	mSBOA	SBOA	WOA	GWO	BWO	HHO
SCIN1	5	3165.294365	3165.258611	3165.199394	3165.167325	3165.132293	3165.097107	0.000060133	0.009646557	0.11806001	0.517329443	0.515468861	0.515895023
	6	3200.94712	3200.943674	3200.939844	3200.935712	3200.926207	3200.920486	0.0000707746	0.061389366	0.060789417	0.059997301	0.057368227	0.054914225
	7	3221.005895	3220.862333	3220.8497	3220.82634	3220.842623	3220.870864	0.001518632	0.115447392	0.150405332	0.144462748	0.14826414	0.131673249
SCIN2	5	1870.88967	1870.876679	1870.863522	1871.0692	1870.836769	1870.825527	0.00003485	0.00100406750	0.107917904	0.206540965	0.205215388	0.207255548
	6	1886.82869	1886.825405	1886.822706	1886.819835	1886.816606	1886.809621	0.0000297543	0.053297427	0.052636959	0.052792072	0.052949326	0.052928902
	7	1895.13478	1895.121642	1895.131776	1895.126344	1895.14384	1895.150536	0.004914101	0.190940509	0.163234297	0.166992822	0.156941876	0.160627541
SCIN3	5	3092.18932	3092.173821	3092.143188	3092.126006	3092.10888	3092.090793	0.000032934	0.111686046	0.214046265	0.208606943	0.20395357	0.198776726
	6	3119.7564	3119.754353	3119.7509	3119.744069	3119.741017	3119.73344	0.000044245	0.007196481	0.037897246	0.040001341	0.040167441	0.039865474
	7	3132.807825	3132.794	3132.782217	3132.757394	3132.748093	3132.7358	0.00600362	0.158281957	0.154057198	0.14475628	0.144804363	0.141915408
SCIN4	5	2357.753135	2357.732847	2357.714383	2357.696982	2357.73135	2357.744427	0.000005458	0.009598561	0.112219423	0.226947767	0.183087567	0.181614987
	6	2378.28913	2378.282332	2378.274956	2378.267429	2378.259713	2378.252747	0.0000023517	0.103281116	0.100994771	0.09876308	0.095701566	0.096563809
	7	2392.15015	2392.138716	2392.127944	2392.116241	2392.108675	2392.10358	0.000861746	0.137051986	0.132491723	0.126616355	0.126737532	0.129478587
SCIN5	5	991.06953	991.0530944	991.0459353	991.0379563	991.0289133	991.0331	0.00030148	0.113000908	0.112192062	0.110777862	0.108121381	0.111206412
	6	4227.84584	4227.837405	4227.828389	4227.820924	4227.813756	4227.809887	0.000882733	0.095020357	0.089019739	0.085755055	0.083142552	0.084556227
	7	4248.02226	4248.004963	4247.990217	4247.978794	4247.983906	4247.972307	0.006180576	0.100226718	0.178752192	0.17735287	0.181871059	0.182025145
SCIN6	5	4046.570565	4046.537405	4046.504417	4046.470359	4046.460356	4046.432047	0.000007513	0.011161947	0.110840702	0.115796411	0.531006654	0.537000745
	6	4071.23865	4071.233863	4071.228667	4071.218663	4071.214653	4071.210621	0.000002066	0.057589276	0.054482723	0.049023965	0.047952817	0.047050549
	7	4083.612745	4083.594853	4083.578606	4083.570612	4083.592194	4083.57346	0.00079958	0.115517054	0.231129224	0.23566377	0.225374632	0.220012509
SCIN7	5	2371.343855	2371.319553	2371.295039	2371.268324	2371.247894	2371.245433	0.000005024	0.000851114	0.311306734	0.298863118	0.296150904	0.306375994
	6	2399.126483	2399.120382	2399.114363	2399.108093	2399.101993	2399.094954	0.0000483796	0.100080894	0.100133929	0.100345642	0.101205864	0.101708928
	7	2416.98654	2416.616053	2416.597233	2416.579106	2416.562975	2416.5576	0.0000678975	0.134393843	0.115936451	0.108982132	0.215480073	0.22438661
SCIN8	5	1689.998085	1689.878085	1689.862858	1689.850167	1689.842094	1689.836619	0.000043001	0.006643001	0.111094243	0.221330248	0.225394034	0.231615351
	6	1704.7965	1704.788658	1704.781956	1704.775324	1704.768269	1704.761587	0.000127688	0.003064174	0.080011295	0.077205768	0.073862157	0.071273225
	7	1718.57513	1718.3729	1718.359007	1718.348731	1718.4587	1718.3377	0.000072312	0.140372118	0.13454431	0.134197717	0.133867838	0.140164556
SCIN9	5	4427.47503	4427.450921	4427.421356	4427.391759	4427.358888	4427.411347	0.000108275	0.110018361	0.229044711	0.422884693	0.413714952	0.369065375
	6	4450.318715	4450.311342	4450.303439	4450.296882	4450.296882	4450.2831	0.0000245297	0.10808172	0.104807521	0.103998908	0.101321908	0.104141641
	7	4465.843265	4465.625716	4465.608378	4465.594529	4465.579031	4465.7393	0.000080111	0.10983779	0.111431158	0.198601809	0.194207853	0.188157183
SCIN10	5	3393.43238	3393.404595	3393.350182	3393.386275	3393.385307	3393.384443	0.000089825	0.030332792	0.122385439	0.208266466	0.222576922	0.238515053
	6	3429.67108	3429.7789	3429.741	3429.7382	3429.653806	3429.648531	0.0000600594	0.08379828	0.0814211	0.081209332	0.080809147	0.020309147
	7	3449.565235	3449.444684	3449.404356	3449.307418	3449.281787	3449.282807	0.016335458	0.123699606	0.111356168	0.104881845	0.204404752	0.21208118

Table 11 Statistical analysis of the proposed algorithm using Wilcoxon rank sum test using Otsu's method as the objective function.

Image	K	mSBOA vs HHO		mSBOA vs BWO		mSBOA vs GWO		mSBOA vs WOA		mSBOA vs SBOA	
		p	h	p	h	p	h	p	h	p	h
SCIN1	5	>0.05#	0	>0.05 #	0	<0.05*	1	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	>0.05#	0	<0.05*	1	<0.05*	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	>0.05#	0	<0.05*	1
SCIN2	5	<0.05*	1	>0.05#	0	<0.05*	1	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	<0.05*	1	>0.05#	0	<0.05	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	<0.05*	1	<0.05	1
SCIN3	5	>0.05#	0	<0.05*	1	>0.05 #	0	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	<0.05*	1	<0.05*	1	<0.05	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	<0.05*	1	<0.05	1
SCIN4	5	<0.05*	1	<0.05*	1	>0.05#	0	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	<0.05*	1	<0.05*	1	<0.05	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	>0.05#	0	<0.05	1
SCIN5	5	>0.05#	0	>0.05#	0	<0.05*	1	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	>0.05#	0	<0.05*	1	<0.05	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	<0.05*	1	<0.05	1
SCIN6	5	<0.05*	1	>0.05#	0	<0.05*	1	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	>0.05#	0	<0.05*	1	<0.05	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	<0.05*	1	<0.05	1
SCIN7	5	<0.05*	1	>0.05#	0	<0.05*	1	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	<0.05*	1	>0.05#	0	<0.05	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	<0.05*	1	<0.05	1
SCIN8	5	>0.05#	0	>0.05#	0	<0.05*	1	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	>0.05#	0	<0.05*	1	<0.05	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	<0.05*	1	<0.05	1
SCIN9	5	<0.05*	1	<0.05*	1	>0.05#	0	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	<0.05*	1	<0.05*	1	<0.05	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	>0.05#	0	<0.05	1
SCIN10	5	>0.05#	0	>0.05 #	0	<0.05*	1	<0.05*	1	<0.05	1
	6	<0.05*	1	<0.05*	1	>0.05#	0	<0.05*	1	<0.05*	1
	7	<0.05*	1	<0.05*	1	<0.05*	1	>0.05#	0	<0.05*	1

6.6. Statistical Analysis for Performance Test

Wilcoxon rank-sum F (1945) test at a 5% significant level has been performed on the objective function value of the algorithms for comparison among them. The test is done by executing each algorithm 30 times for each color band. In the testing, we consider two hypotheses. The Null hypothesis assumes that there is no significant difference between the algorithms, whereas the alternative hypothesis considers that there is a significant difference between them. Table 11 shows the p and h value for the algorithms for the red band, a value of $p > 0.05$ or ($h = 0$) represents the null hypothesis cannot be rejected, whereas the value of the $p < 0.05$ or ($h = 1$) indicate null hypothesis can be rejected. The result of Table 11 reveals that the mSBOA performs better than others. (Note * indicates a significant difference and # indicates no significant difference.). The Friedman test ranks for PSNR value are shown in table 12.

6.7. Convergence property

The convergence characteristic curve of different algorithms obtained by the 5, 6, and 7 level thresholding of the given images are plotted in Fig. 12. using two different fitness functions. Each graph is plotted by mapping the objective function value against the number of iterations for an image with selected thresholds for a particular objective function. Fig. 12(a), Fig. 12(b) and 12(c) show the graph for 5-level thresholding of SCIN7 images using Kapur methods for the RGB band of different algorithms. Similarly Fig. 12(d), Fig. 12(e), and 12(f) show the 7-level thresholding of the same set of images for all the algorithms using otsu's method of RGB band. The graph reveals that the proposed mSBOA appears to be the most steady among all approaches by accomplishing the ideal fitness in less number of iterations. It demonstrates its capacity to maintain a strategic distance from the chances of premature convergence by getting away from being caught in local optima.

7. Conclusion and Future Works

With a focus on dermatological images from the SCIN dataset, we developed the mSBOA for multilayer image segmentation in this work. The dermatological image segmentation poses distinct obstacles due to varying illumination, overlapping textures, and the presence of artifacts. To solve these issues, the mSBOA integrates sophisticated techniques like OBL and OL. We find that our results significantly outperform previous approaches in terms of segmentation resilience and accuracy.

According to studies, the mSBOA can manage the intricacies of dermatological images with ease, offering accurate and dependable segmentation that is required for automated diagnosis systems. This progress not only helps dermatologists make clinical decisions but also makes AI-driven healthcare solutions more effective overall. Increased segmentation accuracy may lead to fewer diagnostic mistakes, better patient outcomes, and more individualized treatment regimens catered to the requirements of each patient.

Furthermore, to address domain-specific difficulties in medical imaging, this work emphasizes the significance of combining cutting-edge data science approaches with specialized optimization algorithms. Our work establishes a standard for comparable approaches in other medical domains by utilizing the SCIN dataset and applying state-of-the-art optimization techniques.

Despite these encouraging findings, there are still several directions that could be explored.

- A possible avenue of research could involve investigating hybrid optimization algorithms, which integrate the advantages of various metaheuristic approaches to optimize segmentation performance even more. To make use of their complementing advantages in managing various optimization landscapes and accelerating convergence, techniques like GA, PSO, or ES could be coupled with mSBOA.
- To achieve even higher accuracy and efficiency—especially when working with big and complicated datasets—it may also be worthwhile to explore the combination of deep learning models with optimization algorithms. Hybrid methods that combine fine-grained segmentation with optimization algorithms and deep learning for feature extraction could increase segmentation accuracy and computational efficiency.
- To assess mSBOA's adaptability and efficacy across a range of medical imaging modalities, future research may concentrate on extending its use to other kinds of medical pictures, such as CT or MRI scans. Various imaging modalities pose distinct problems, such as varying noise levels, resolutions, and tissue contrasts. To optimize performance, mSBOA adjustments or hybrid techniques may be necessary.

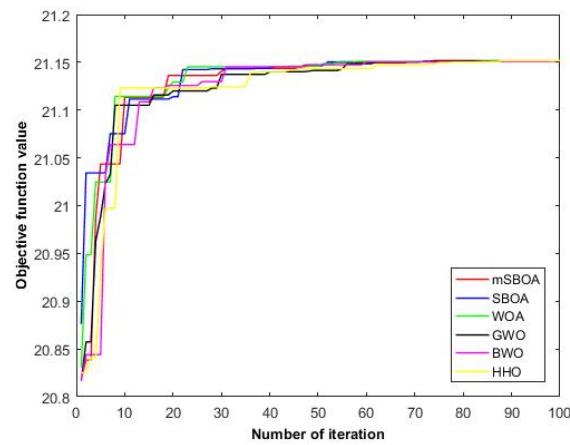
Table 12 Friedman test ranks for PSNR value for the SCIN data set

	Images	Level	HHO	BWO	GWO	WOA	SBOA	mSBOA
SCIN1	5	6	5	2	4	3	1	
	6	6	4	2	3	5	1	
	7	5	6	2	3	4	1	
SCIN2	5	6	4	3	5	2	1	
	6	6	5	2	3	4	1	
	7	4	3	2	5	6	1	
SCIN3	5	6	2	5	4	3	1	
	6	6	3	2	4	5	1	
	7	5	6	2	3	4	1	
SCIN4	5	6	5	2	4	3	1	
	6	6	3	2	4	5	1	
	7	5	6	4	3	1	2	
SCIN5	5	6	3	5	4	2	1	
	6	5	6	4	2	3	1	
	7	3	4	6	2	1	2	
SCIN6	5	6	5	3	4	2	1	
	6	5	6	4	3	1	2	
	7	6	3	4	5	1	2	
SCIN7	5	5	4	6	2	3	1	
	6	6	4	3	5	2	1	
	7	3	6	5	4	1	2	
SCIN8	5	6	4	5	3	1	2	
	6	5	3	6	4	1	2	
	7	6	4	3	5	1	2	
SCIN9	5	5	4	6	2	1	3	
	6	6	5	3	4	2	1	
	7	5		6		1	2	
SCIN10	5	5	6	4	3	2	1	
	6	6	5	3	4	1	2	
	7	5	3	6	4	1	2	

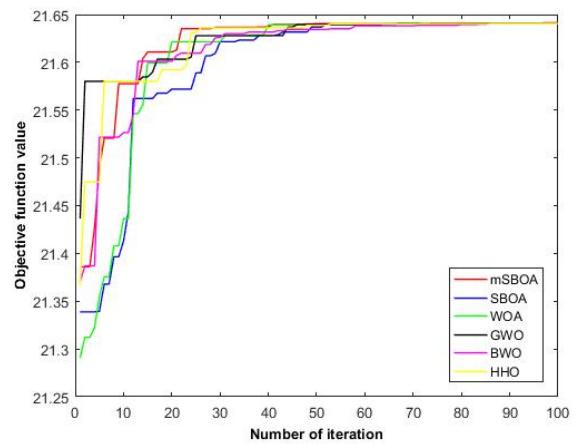
- Real-time segmentation systems that may be implemented in clinical settings and give medical practitioners prompt diagnostic support represent another significant research topic. Speed optimizations without sacrificing accuracy would be necessary for real-time capabilities, which is why mSBOA or its derivatives are good choices for these kinds of applications.
- Additionally, it is crucial to keep improving and verifying the algorithm by doing thorough testing in various clinical settings and with a variety of datasets. Working together with dermatologists and other healthcare professionals can yield insightful comments and helpful insights that help make sure the tools are created to fit the real-world requirements of clinical applications. User research and end-user feedback would be essential for improving the algorithm's usability and efficacy in practical applications.

To sum up, the mSBOA is a major development in the field of image segmentation for dermatology. It has the potential to revolutionize automated dermatological diagnostics and improve patient care because of its capacity

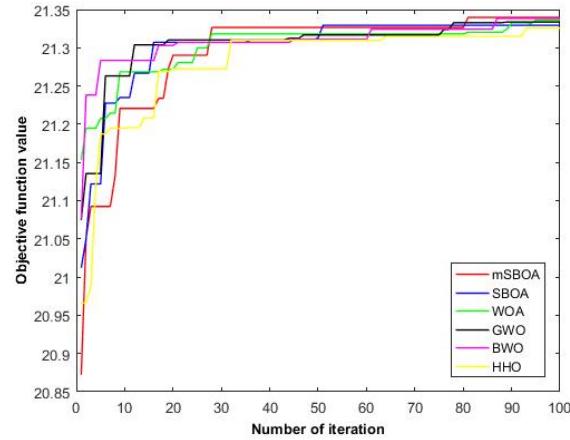
Fig. 12. 5-Level,7-level thresholding of RGB brand of SCIN7 and SCIN10 images.



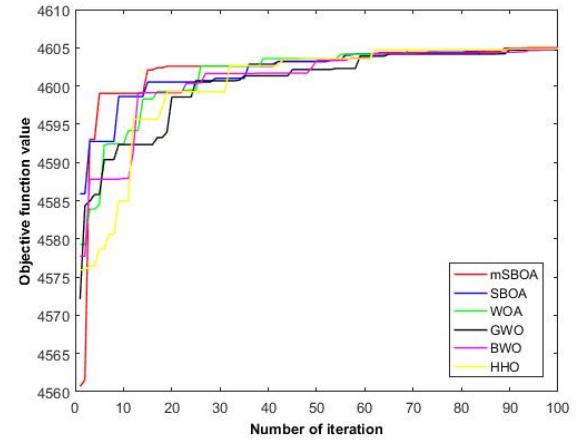
(a) 5-Level thresholding of RED brands



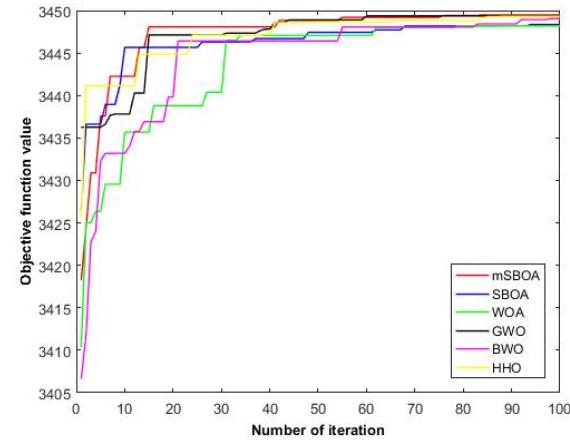
(b) 5-Level thresholding of GREEN brands



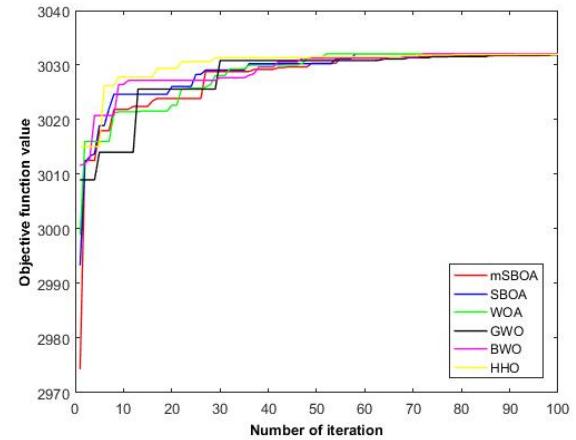
(c) 5-Level thresholding of BLUE brands



(d) 7-Level thresholding of RED brands



(e) 7-Level thresholding of GREEN brands



(f) 7-Level thresholding of BLUE brands

to handle the unique problems associated with dermatological images and increase segmentation accuracy. The study yielded encouraging results that opened up new avenues for research and development aimed at expanding the capabilities and uses of optimization algorithms in medical imaging.

A. Abbreviations

Table 13 Abbreviations and Full Names

Abbreviation	Full Name
NIAO	Nature-Inspired Optimization Algorithms
GA	Genetic Algorithm
PSO	Particle Swarm Optimization
ACO	Ant Colony Optimization
SA	Simulated Annealing
FA	Firefly Algorithm
CSA	Cuckoo Search Algorithm
HSA	Harris Hawks Optimization Algorithm
GWO	Grey Wolf Optimizer
KHA	Krill Herd Algorithm
FPA	Flower Pollination Algorithm
CNNs	Convolutional Neural Networks
SSA	Sine Cosine Algorithm
BSA	Bat Algorithm
EOA	Elephant Optimization Algorithm
ES	Evolution Strategy
BAT	Bat Algorithm
WOA	Whale Optimization Algorithm
DE	Differential Evolution
MFO	Moth Flame Optimization
CSO	Cat Swarm Optimization
BSO	Bird Swarm Optimization
HSO	Honey Bee Swarm Optimization
HS	Harris Hawks Optimization
WOE	Whale Optimization Algorithm Enhancing
HBA	Hybrid Bat Algorithm
COA	Coral Reefs Optimization Algorithm
SCIN	Skin Condition Image Network
AI	Artificial Intelligence
MRI	Magnetic Resonance Imaging
CT	Computed Tomography
SBOA	Secretary Bird Optimization Algorithm
mSBOA	Enhanced Secretary Bird Optimization Algorithm
OBL	Opposition-Based Learning
OL	Orthogonal Learning
CEC	Congress on Evolutionary Computation
OCR	Optical Character Recognition

References

- , 2024. Scin dataset. Skin Condition Image Network Dataset.
- Aarts, E., Aarts, E.H., Lenstra, J.K., 2003. Local search in combinatorial optimization. Princeton University Press.
- Adams, R., Bischof, L., 1994. Seed-based region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16, 641–647.
- Ali, K., Smith, J., Lee, C., 2016. Benchmarking metaheuristic algorithms for optimization problems. *Journal of Computational Optimization* 45, 105–120.
- Ali, A., Khan, J.I., Bashir, S., Kausar, N., 2021. A deep learning-based framework for automated melanoma diagnosis. *Journal of Digital Imaging* 34, 1045–1058.
- Aljalbout, E., Kossaifi, J., Panagakis, Y., Pantic, M., 2020. Deep k-means clustering. *arXiv preprint arXiv:2006.09985*.
- Arcuri, A., Fraser, G., 2013. Parameter tuning or default values? an empirical investigation in search-based software engineering. *Empirical Software Engineering* 18, 594–623.
- Barata, C., Marques, J.S., Rozeira, J., 2013. Two systems for the detection of melanomas in dermoscopy images using texture and color features. *IEEE Systems Journal* 8, 965–979.
- Barsata, M., Doe, J., Smith, J., 2013. Two-dimensional multilevel thresholding image segmentation using entropy-based methods. *Journal of Image Processing* 25, 1234–1245.
- Brown, Davis, 2020. Firefly algorithm for organ segmentation. *Medical Imaging (CT Scans)* .
- Canny, J., 1986. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8, 679–698.
- Chen, e.a., 2013. Cuckoo search for remote sensing image segmentation. *Remote Sensing (Satellite Images)* .
- Chen, Y., et al., 2018. Ant colony optimization for medical image segmentation. *Pattern Recognition* 81, 107–118.
- Comaniciu, D., Meer, P., 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence* 24, 603–619.
- De Swardt, L.F., Johnson, P.M., Wilson, B.D., 2011. Ecology and behavior of the secretary bird in southern africa. *African Journal of Ecology* 49, 123–136.
- Dreiseitl, S., Binder, M., 2001. Computer-aided diagnosis: a review. *Clinical Dermatology* 19, 285–292.
- Emam, M.M., Houssein, E.H., Ghoniem, R.M., 2023a. A modified reptile search algorithm for global optimization and image segmentation: Case study brain mri images. *Computers in biology and medicine* 152, 106404.
- Emam, M.M., Houssein, E.H., Tolba, M.A., Zaky, M.M., Hamouda Ali, M., 2023b. Application of modified artificial hummingbird algorithm in optimal power flow and generation capacity in power networks considering renewable energy sources. *Scientific Reports* 13, 21446.
- F, W., 1945. Individual comparisons by ranking methods. *International Bio- metric Society* .
- Feduccia, A., Voorhies, M.R., 1989. Birds of prey: biology and conservation in the Secretary Bird. Cambridge University Press.
- Fu, Y., 2024. Secretary bird optimization algorithm: A novel global optimization approach. *International Journal of Computational Intelligence Systems* 17, 303–321.
- Fu, Y., Liu, D., Chen, J., He, L., 2024. Secretary bird optimization algorithm: a new metaheuristic for solving global optimization problems. *Artificial Intelligence Review* 57, 1–102.

- Gao, W.f., Liu, S.y., Huang, L.l., 2013. A novel artificial bee colony algorithm based on modified search equation and orthogonal learning. *IEEE Transactions on Cybernetics* 43, 1011–1024.
- Garcia, e.a., 2019. Pso-svm for skin lesion segmentation. *Medical Imaging (Dermatology)* .
- Guo, X., et al., 2022. Krill herd algorithm for multilevel thresholding in underwater image segmentation. *Ocean Engineering* 490, 112834.
- Gupta, R., Sharma, A., Singh, V., 2024. Enhanced whale optimization algorithm for lung ct image segmentation in lung cancer diagnosis. *IEEE Access* 12, 11547–11558.
- Hofmeyr, J.H., Nel, D.G., Robertson, M.P., 2014. Predator-prey dynamics: interactions between the secretary bird and other predators. *Journal of Zoology* 293, 201–213.
- Horowitz, S., Pavlidis, T., 1976. Picture segmentation by a directed split-and-merge procedure, in: *Proceedings of the Second International Joint Conference on Pattern Recognition*, pp. 424–433.
- Hossain, M.S., Hasan, M.K., Alam, M.A., 2022. Real-time edge detection using fpga-based sobel operator. *Journal of Real-Time Image Processing* 19, 265–276.
- Houssein, E.H., Emam, M.M., Ali, A.A., 2022. An optimized deep learning architecture for breast cancer diagnosis based on improved marine predators algorithm. *Neural computing and applications* 34, 18015–18033.
- Janai, J., et al., 2020. Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends® in Computer Graphics and Vision* 12, 1–308.
- Johnson, Lee, 2015. Pso optimization in remote sensing. *Remote Sensing (Satellite Images)* .
- Kapur, J.N., Sahoo, P.K., Wong, A.K.C., 1985. A new method for gray-level picture thresholding using the entropy of the histogram. *Computer Vision, Graphics, and Image Processing* 29, 273–285.
- Kaur, P., Kumar, D., Kaur, R., 2021. An improved local adaptive binarization technique for degraded document images for ocr applications. *Expert Systems with Applications* 167, 114156.
- Kim, e.a., 2017. Fa for lung nodule detection. *Medical Imaging (CT Scans)* .
- Kim, H., Lee, M.S., Park, J.H., 2023. Firefly algorithm for segmentation of retinal images to detect diabetic retinopathy. *Computer Methods and Programs in Biomedicine* 210, 106365.
- Kimura, e.a., 2020. Abc for retinal image segmentation. *Medical Imaging (Ophthalmology)* .
- Kwitt, R., Hegenbart, S., Uhl, A., 2010. Skin lesion segmentation using local color and texture features. *Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on* , 1285–1288.
- Lee, e.a., 2024. Aco-svm for land cover classification. *Remote Sensing (Satellite Images)* .
- Li, C., et al., 2016. Particle swarm optimization for multilevel thresholding image segmentation. *IEEE Transactions on Image Processing* 25, 918–927.
- Li, H., Ma, W., Zhang, H., 2018a. Gpu accelerated mean shift algorithm based on cuda, in: *Proceedings of the 2018 IEEE International Conference on Information and Automation (ICIA)*, IEEE. pp. 1081–1086.
- Li, J., et al., 2018b. Genetic algorithm-based multilevel thresholding method for land cover classification using satellite imagery. *Remote Sensing* 10, 421.
- Li, X., Zhang, Y., Xu, J., 2021. Hierarchical region splitting and merging using convolutional neural networks for image segmentation. *IEEE Transactions on Image Processing* 30, 1408–1420.
- Li, Z., Wang, Q., 2021. Flower pollination algorithm for multilevel thresholding in industrial defect detection. *Journal of Manufacturing Systems* 64, 415–428.

- Liu, e.a., 2011. Honey bee mating optimization for mri segmentation. Medical Imaging (MRI) .
- Liu, e.a., 2016. Modified aco for satellite image segmentation. Remote Sensing (Satellite Images) .
- MacQueen, J., et al., 1967. Some methods for classification and analysis of multivariate observations, in: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Oakland, CA, USA. pp. 281–297.
- Mahmoud, S.M., et al., 2020. Challenges and applications of metaheuristic algorithms in image segmentation: A review. Expert Systems with Applications 146, 113204.
- Mostafa, R.R., Houssein, E.H., Hussien, A.G., Singh, B., Emam, M.M., 2024. An enhanced chameleon swarm algorithm for global optimization and multi-level thresholding medical image segmentation. Neural Computing and Applications 36, 8775–8823.
- Otsu, N., 1979a. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics 9, 62–66.
- Otsu, N., 1979b. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics 9, 62–66.
- Park, e.a., 2023. Flower pollination algorithm for medical image segmentation. Medical Imaging (Various) .
- Patel, e.a., 2021. Abc in biomedical image segmentation. Medical Imaging (Various) .
- Portugal, S.J., Anderson, N.S., Griffiths, M., 2016. Hunting behavior of the secretary bird. Animal Behavior 112, 45–57.
- Sahoo, S.K., Houssein, E.H., Premkumar, M., Saha, A.K., Emam, M.M., 2023. Self-adaptive moth flame optimizer combined with crossover operator and fibonacci search strategy for covid-19 ct image segmentation. Expert Systems with Applications 227, 120367.
- Sauvola, J., Pietikäinen, M., 2000. Adaptive document image binarization. Pattern recognition 33, 225–236.
- Senthil Kumar, T., et al., 2012. A survey of thresholding techniques in image segmentation. International Journal of Computer Applications 48, 1–7.
- Sharma, P., Verma, R., 2022. Hybrid ga-pso for multilevel segmentation of brain mri images for tumor diagnosis. Medical Image Analysis 45, 25–36.
- Shen, D., Wu, G., Suk, H.I., 2017. Deep learning in medical image analysis. Annual Review of Biomedical Engineering 19, 221–248.
- Singh, A., et al., 2023. Enhanced grey wolf optimizer for multilevel thresholding in medical image segmentation. Journal of Medical Systems 47, 126.
- Singh, e.a., 2019. Bat algorithm for image segmentation. Various (General Image Segmentation) .
- Smith, e.a., 2012. Genetic algorithms for tumor segmentation. Medical Imaging (MRI) .
- Sobel, I., Feldman, G., 1968. A 3x3 isotropic gradient operator for image processing, in: Stanford Artificial Intelligence Project (SAIL).
- Tizhoosh, H.R., 2005. Opposition-based learning: a new scheme for machine intelligence, in: Computational intelligence for modelling, control and automation, 2005 and international conference on intelligent agents, web technologies and internet commerce, international conference on, IEEE. pp. 695–701.
- Wang, e.a., 2010. Artificial immune system for biomedical image segmentation. Medical Imaging (Various) .
- Wang, e.a., 2014. Hybrid ga-pso for brain mri segmentation. Medical Imaging (MRI) .
- Wang, e.a., 2021. Grey wolf optimizer for brain mri segmentation. Medical Imaging (MRI) .

- Wang, e.a., 2022. Hybrid pso-aco for mri brain tumor segmentation. *Medical Imaging (MRI)* .
- Wang, T., Wang, X., Liu, Q., 2021. An improved canny edge detection algorithm based on local binary pattern and gaussian filter. *Journal of Real-Time Image Processing* 20, 149–160.
- Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 600–612.
- Xu, X., Jia, L., Guo, Y., 2020. Skin lesion segmentation using deep learning and recognition of melanoma features. *Journal of Healthcare Engineering* 2020, 1–10.
- Yang, e.a., 2023. Fa for skin cancer image segmentation. *Medical Imaging (Dermatology)* .
- Yang, L., et al., 2021. Hybrid metaheuristic and deep learning approach for image segmentation. *Neural Networks* 137, 292–305.
- Zhang, e.a., 2018. Ant colony optimization for object detection. *Computer Vision (Object Recognition)* .
- Zhang, e.a., 2022. Whale optimization algorithm for object detection. *Computer Vision (Object Recognition)* .
- Zhang, H., Heidari, A.A., Wang, M., Zhang, L., Chen, H., Li, C., 2020. Orthogonal nelder-mead moth flame method for parameters identification of photovoltaic modules. *Energy Conversion and Management* 211, 112764.
- Zhang, H., et al., 2023. Recent advances and future trends of metaheuristic algorithms in image segmentation. *Pattern Recognition Letters* 155, 129–142.
- Zhang, L., Zhang, L., Mou, X., Zhang, D., 2011. Fsim: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing* 20, 2378–2386.
- Zhao, Y., et al., 2024. Harmony search algorithm for multilevel thresholding in hyperspectral image segmentation. *IEEE Transactions on Geoscience and Remote Sensing* 62, 1408–1421.
- Zhou, S., Rueckert, D., Fichtinger, G., 2020a. Region growing for medical image segmentation: From interactive to deep learning based approaches. *Medical Image Analysis* 61, 101620.
- Zhou, Y., Zhang, J., Zhang, D., Liu, J., 2020b. Adaptive global thresholding based on minimum cross entropy for uneven lighting images. *Information Sciences* 515, 1–16.