IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 9 February 2024, accepted 14 February 2024, date of publication 22 February 2024, date of current version 29 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3369039

RESEARCH ARTICLE

Optimizing Road Traffic Surveillance: A Robust Hyper-Heuristic Approach for Vehicle Segmentation

ERICK RODRÍGUEZ-ESPARZA[1], OSCAR RAMOS-SOTO[2], ANTONIO D. MASEGOSA[1,3], ENRIQUE ONIEVA[1], DIEGO OLIVA[2], (Senior Member, IEEE), ANDER ARRIANDIAGA[1], AND ARKA GHOSH[1]

[1]DeustoTech, Faculty of Engineering, University of Deusto, 48007 Bilbao, Spain
[2]Departamento de Ingeniería Electro-Fotónica, CUCEI, Universidad de Guadalajara, Guadalajara, Jalisco 44430, Mexico
[3]Ikerbasque, Basque Foundation for Science, 48009 Bilbao, Spain

Corresponding author: Erick Rodríguez-Esparza (erick.rodriguez@deusto.es)

This work was supported in part by the University of Deusto Research Training Grants Programme through Spanish Ministry of Science and Innovation under Project PID2022-140612OB-I00; and in part by the Basque Government under Grant IT1564-22, Grant KK-2023/00012, and Grant KK-2023/00038.

ABSTRACT Due to rising consumer demand and traffic congestion, last-mile logistics is becoming more challenging. To optimize urban distribution networks, digital image processing plays a key role in addressing these challenges through efficient traffic monitoring systems, an essential component of intelligent transportation systems. This paper introduces the Hyper-heuristic Genetic Algorithm based on Thompson Sampling with Diversity (HHGATSD), a novel approach to efficiently solving complex optimization and versatility problems in image segmentation. We evaluate its efficiency and robustness using the IEEE CEC2017 benchmark function set in general optimization problems with 30 and 50 dimensions. HHGATSD's applicability extends beyond optimization to computer vision in traffic management. First, the multilevel thresholding segmentation is performed on images extracted from the Berkeley Segmentation Dataset with minimum cross-entropy as the objective function, and its performance is compared using PSNR, SSIM, and FSIM metrics. Following that, the proposed methodology addresses the task of vehicle segmentation in traffic camera videos, reaffirming HHGATSD's effectiveness, adaptability, and consistency by consistently outperforming alternative segmentation methods found in the state-of-the-art. The results of comprehensive experiments, validated by statistical and non-parametric analyses, show that the proposed hyper-heuristic and methodology produce accurate and consistent segmentations for road traffic surveillance compared to the other methods in the literature.

INDEX TERMS Digital image processing, hyper-heuristic optimization, intelligent transportation systems, multilevel thresholding, traffic surveillance, vehicle segmentation.

I. INTRODUCTION

The relentless march of technological advancement has revolutionized numerous facets of our lives, none more so than the e-commerce industry. As a result, online shopping has undergone a profound transformation, ushering in an era of unprecedented convenience for consumers. However, this convenience has given rise to a complex logistical challenge known as the "last-mile logistics" [1]. This concept represents the final leg of the distribution process, the critical juncture at which goods reach the end client's doorstep. In this entangled web of distribution, multiple stops and diverse modes of transportation are often involved in fulfilling a single delivery request; such complexity invariably drives up the operational costs for logistics companies.

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Sharif.

Before a product reaches its intended destination, several factors must be carefully considered: the selection of optimal routes, cost constraints, the choice of transportation modes, and traffic bottlenecks, among others [2]. Logistics companies have high stakes, as customer satisfaction hinges on damage-free and on-time deliveries. Efficient last-mile logistics necessitate a delicate balance of optimized routes, minimal traffic congestion, and reduced reliance on personnel; elements focused on diminishing transportation costs.

Despite the considerable attention that last-mile logistics has aroused in the research area [3], [4], its challenges have grown in tandem with technological advances and consumer expectations [5], [6]. A significant factor in urban distribution networks is traffic congestion, an inescapable reality that can have far-reaching consequences for logistics operations [7]. The time spent delivering congested thoroughfares represents a heavy toll on logistics companies, impacting their bottom line, delivery schedules, and the feasibility of meeting allocated time frames.

In this context, the importance of effective traffic management, often referred to as the main issue of urban logistics [8], difficult the efficient flow of goods and significantly inflates operational expenses, representing lost margins for logistics companies. Traffic image analysis is the main approach of modern traffic management systems, offering a smart approach to deal with complex challenges caused by congested roadways. This step enhances the overall efficiency of traffic management and provides a crucial basis for subsequent analyses and decision-making. Through examination of road traffic surveillance camera feeds, image analysis becomes the eyes and brains of traffic management systems, tirelessly scrutinizing every intersection, highway, and arterial road. Doing this offers invaluable support in the endeavor to untangle the web of urban congestion, promoting smoother logistics operations and safer and more sustainable urban environments.

This article presents an innovative hyper-heuristic (HH) approach for traffic image analysis, offering an alternative solution for traffic management systems improvement. By leveraging the Hyper-heuristic Genetic Algorithm Based on Thompson Sampling with Diversity (HHGATSD), this research is intended to provide logistics companies an alternative to decode and analyze complex traffic scenarios efficiently. The work presented in this paper holds excellent promise as an indispensable pre-tool for traffic monitoring within the logistics domain.

The article's key contributions can be outlined as follows:

- **Efficient Hyper-Heuristic:** The proposed HHGATSD excels on the IEEE CEC2017 benchmark, providing a potent algorithm for complex optimization, and it showcases its versatility by achieving outstanding results in image segmentation.
- **Precision Vehicle Detection:** This work introduces a novel and efficient method for vehicle segmentation without any training process or prior knowledge by utilizing the minimum cross-entropy as the objective function, which addresses a critical aspect of last-mile logistics and traffic management.
- **Adaptive Image Segmentation:** In addition to vehicle detection, the paper proposes an adaptive image segmentation technique, contributing to the broader computer vision and pattern recognition field.
- **Traffic Surveillance:** The methodology presented in this research extends beyond traditional image analysis, encompassing traffic surveillance capabilities to assist in urban planning and enhancing logistics operations by providing insights into traffic dynamics.

The remainder of the paper is distributed as follows: Section II includes a State-of-the-art analysis of metaheuristics (MH) and HH algorithms and traffic image analysis. Section III briefly explains the cross-entropy minimization as an image segmentation criterion. Section IV unveils the HHGATSD and its design considerations, while in Section V the experimental framework for the HHGATSD test and the methodology for vehicle segmentation. Section VI shows the performance results, including experiments with three different optimization scenarios. Finally, in Section VII, the conclusions of this work are presented.

II. STATE-OF-THE-ART

In the present section, we embark on a brief but comprehensive review of MH and HH algorithms and their applications, as well as an exploration of cutting-edge techniques in the realm of traffic image analysis.

A. METAHEURISTIC AND HYPER-HEURISTIC ALGORITHMS

Image segmentation, a fundamental process in computer vision and image processing, divides an image into distinct and meaningful regions based on visual properties like color, intensity, and texture. Approaches include region-based, edge-based clustering (e.g., K-means), deep learning, and both MH and HH methods. This section focuses on MH and HH in its research.

Generally, MH algorithms are a class of computational methods used to find approximate solutions to optimization and search problems [9]. These algorithms are particularly valuable for solving complex problems where finding an exact optimal solution within a reasonable amount of time is challenging. Classical MH algorithms draw inspiration from diverse principles and natural phenomena, leading to their categorization into different families.

These algorithms draw inspiration from diverse sources. Swarm-based algorithms mimic collective behaviors, including Particle Swarm Optimization (PSO) [10], Ant Colony Optimization, Grey Wolf Optimizer (GWO) [11], and Side-Blotched Lizard Algorithm [12]. Evolutionary-based algorithms, such as Genetic Algorithm (GA) [13], Differential Evolution (DE), and Self-adaptive Differential Evolution (JADE) [14], emulate natural selection. Physics-based

algorithms, like Simulated Annealing, Harmony Search, and Gravitational Search Algorithm, model physical phenomena.

As the ever-evolving optimization research area has advanced, new-generation and high-performance MH algorithms have emerged, such as Bernstain-Search Differential Evolution Algorithm (BSDE) [15], Biogeography-based Learning Particle Swarm Optimization [16], Harris Hawks Optimization (HHO) [17], Remora Optimization Algorithm [18], to cite a few; even metaphor-free proposals have been developed [19]. The MH approach for image segmentation has been successfully applied to different areas, comprising industrial failure detection [20], medical applications [21], [22], forest management [23], satellite imagery [24], hyperspectral data analysis [25], to mention some.

Nevertheless, applying existing algorithms in certain areas remains challenging due to numerous parameters and their susceptibility to configuration intricacies, as noted in previous research [26]. Consequently, selecting an appropriate search mechanism is paramount as it is pivotal in achieving precise and optimal outcomes. In this context, the adoption of HH has garnered increasing attention. HH represents a class of sophisticated automated search techniques that effectively amalgamate, generate and choose low-level heuristics, thereby offering a powerful means to address computationally intricate problems [27], [28]. While MH is tailored to specific classes of problems and relies on predefined algorithms and low-level heuristics, HH aims to automate the process of selecting or generating low-level heuristics, making them versatile and problem-agnostic [29].

Among the most popular HH algorithms are Choice Function HH [30] and Great Deluge HH [31]. Choice Function HH leverages a choice function to select from a predefined set of low-level heuristics, adapting and learning over time to enhance its optimization performance. On the other hand, inspired by the Great Deluge algorithm, the Great Deluge HH progressively reduces acceptance thresholds, iteratively applying low-level heuristics to converge toward optimal solutions. Memetic HH [27] algorithms combine the concepts of memetic algorithms and HH, using populations and iterative application of heuristics for solution refinement. Borrowing from the multi-armed bandit problem, Multi-Armed Bandit HH [32] algorithms balance exploring different heuristics with exploiting the most promising ones. Also, a framework that stands out for designing and comparing HH is the HyFlex [33]; rather than a specific HH algorithm analysis, HyFlex provides a platform to develop and test various HH approaches on different combinatorial optimization problems. These popular HH approaches are pivotal in automating heuristic selection, enabling efficient optimization across many problem domains.

HH algorithms have been applied in a wide range of areas, including assessment of combinatorial optimization problems [34], metallic particles detection [35], [36], supply chain management [37], and improvement of HH algorithms [38], including image segmentation [39], [40]. Lying in the high-effectiveness of HH approaches, different traffic management proposals have been performed, to name a few, scheduling [41], lane-changing advisory [42], public transport network optimization [43], vehicle routing problem [44], and capacitated electric vehicle routing problem [45].

B. IMAGE VEHICLE DETECTION

Regarding vehicle segmentation on road images, several proposals have been presented with different approaches, each offering a distinct and innovative task analysis.

In their 2019 study, Prakoso et al. conducted a comprehensive study of Otsu's thresholding, Fuzzy C-means, and K-means applied to video frames. They assessed the efficacy of these techniques by employing two performance metrics, Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR), as outlined in their work [46]. Later in 2020, Premachandra et al. [47] introduced an innovative method for identifying moving street objects (such as vehicles, pedestrians, cyclists, etc) utilizing the Gaussian mixture model from 360-degree-view images. This represents a noteworthy application of background and frame subtraction techniques [48]. However, these approaches frequently exhibit a reliance on factors such as variations in lighting conditions and background clutter, which can impact their performance and robustness.

In the realm of deep learning, Hu et al. introduced a scale-insensitive convolutional neural network (CNN) for rapid vehicle detection, achieving speeds up to 37 FPS [49]. Similarly, Sindhu utilized YOLOv4 to enhance vehicle detection and identification in CCTV camera video streams [50]. Building on this, Kashevnik et al. proposed a novel approach by combining the EfficientNetB3 architecture with multiparallel residual blocks, aiming at 3D localization and pose estimation of vehicles, inspired by the CenterNet architecture [51]. Despite their efficacy, it is important to highlight that all these deep learning-based methods necessitate a training phase and a substantial number of annotated frames with corresponding vehicle ground truth, presenting challenges regarding computational cost and segmentation time.

Nevertheless, recent advances in deep learning, particularly in few-shot learning, provide promising alternatives. Innovative works, such as the works of Cheng et al. [52] and Lang et al. [53], have introduced schemes that operate effectively with fewer annotated frames. Unlike traditional deep learning methods, these few-shot segmentation approaches work well with less data and offer potential efficiency improvements and reduced computational burden. It is worth noting that while these few-shot segmentation approaches require some training, their unique methodologies contribute to faster adaptation and may mitigate concerns related to computational cost and segmentation time.

Regarding traffic management, a few HH proposals have been presented. In 2022, Khassiba and Delahaye [54] proposed a Simulated-Annealing HH to analyze various delay-based and rerouting-based neighborhood operators.

The optimal combination is then employed to address a demanding and large-scale instance within a relatively brief computational timeframe. More recently, in 2023, Liao et al. [55] tackled the Traffic assignment problem using a genetic programming HH approach for real-time assignment of uncertain commuters. They presented a reactive assignment strategy as a low-level heuristic rule evolved through the proposed method. By training using a designed heuristic template, commuters can dynamically find optimal paths in real time, maximizing traffic network throughput. Also, in 2023, Zheng et al. [56] designed an HH algorithm based on Tabu search, incorporating a high-level heuristic strategy to more efficiently select underlying search operators to optimize vehicle routes by balancing distribution costs and customer satisfaction. This approach is applied to a model considering a time-dependent speed, reflecting vehicle travel speed and road traffic flow changes in urban traffic flow simulations.

To the best of our knowledge, only a limited number of heuristic proposals (including MH and HH approaches) have been specifically designed to segment traffic vehicles. In 2020, Huang et al. [57] presented a hyper-spherical hash algorithm, a high-dimensional heuristic enhancement, to provide vehicle identification for Intelligent traffic analysis. Notable examples include the utilization of the Haversine formula and a recent innovative approach known as the HH-based Encoder-Decoder using Gated Recurrent Units with an attention mechanism, as introduced by Priya et al. in their work presented the current year [58].

III. MULTILEVEL IMAGE SEGMENTATION

Multilevel image segmentation aims to classify pixels with similar attributes, such as intensity, to detect objects within the background. This technique is commonly employed as an initial step in several image processing applications, where it converts the image representation into a more analytically manageable format for further processing stages.

The selection of thresholds is a challenging task in multilevel image segmentation, achievable through diverse manual and automated criteria. This proposal addresses this challenge by employing cross-entropy minimization as a segmentation criterion [59].

The thresholding technique is a simple and widely employed method that involves partitioning the image's histogram into different regions based on specific threshold values. The resulting multilevel segmented image ($\mathbf{I_{th}}$) is generated using the threshold values according to the rules specified in (1).

$$I_{th}(x,y) = \begin{cases} th_1, & \text{if } I_{Gr}(x,y) \le th_1 \\ th_i, & \text{if } th_{i-1} < I_{Gr}(x,y) < th_i, \\ th_{nt}, & \text{if } I_{Gr}(x,y) > th_{nt} \end{cases} \tag{1}$$

where $\mathbf{I_{Gr}}$ is the original grayscale image, nt represents the total number of thresholds, and th_i ($\forall\ i = 2, 3, \ldots, nt - 1$) are one of the optimal threshold values used to segment the image.

The challenge of determining optimal thresholds in images is tackled by formulating the task as the minimization of the statistical cross-entropy criterion. This criterion is the objective function within an optimization algorithm, guiding the search for optimal threshold values.

The cross-entropy criterion measures the homogeneity of the histogram information between the original image and its segmentation.

A. MINIMUM CROSS-ENTROPY

In 1968, Kullback et al. proposed cross-entropy as a metric to measure the divergence between two probability distributions. It is formulated as the minimization of a theoretical data distance represented in (2) to find the optimal threshold value, given probability distributions $\mathbf{P}$=$\{p_1,p_2,\ldots,p_N\}$ and $\mathbf{Q}$=$\{q_1,q_2,\ldots,q_N\}$.

$$CE(\mathbf{P},\mathbf{Q}) = \sum_{i=1}^{N} p_i \log \frac{p_i}{q_i} \tag{2}$$

The Minimum Cross-Entropy Thresholding (MCET) technique determines the optimal threshold by minimizing the statistical criterion of cross-entropy of the input grayscale image and the histogram $h(i)$, $i = 1, 2, \ldots, L$, where L is the number of gray intensities present in the image. The image using a single threshold (th) for binary thresholding, is given by (3) to segment image $\mathbf{I_{Gr}}$ into an $\mathbf{I_{th}}$ image with two areas, foreground, and background.

$$I_{th}(x,y) = \begin{cases} \mu(1, th), & \text{if } I_{Gr}(x,y) < th \\ \mu(th, L+1), & \text{if } I_{Gr}(x,y) \ge th \end{cases} \tag{3}$$

where μ is defined in (4)

$$\mu(a,b) = \sum_{i=a}^{b-1} ih(i) \Big/ \sum_{i=a}^{b-1} h(i) \tag{4}$$

The MCET is calculated by rewriting (3) to obtain the entropy value by evaluating the objective function shown in (5). This function, denoted as $f_{MCET}(th)$, combines information from the histogram, image, and optimal threshold th to measure the divergence between the original and the segmented image.

$$f_{MCET}(th) = \left(\sum_{i=1}^{th-1} ih(i) \log \left(\frac{i}{\mu(1, th)} \right) + \sum_{i=th}^{L} ih(i) \log \left(\frac{i}{\mu(th, L+1)} \right) \right) \tag{5}$$

This bilevel segmentation equation mentioned above can be extended to a multilevel approach using the vector $\mathbf{th} = [th_1, th_2 \ldots, th_{nt}]$, which contains nt threshold values. However, extending to more values is computationally expensive. Nevertheless, Yin [60] proposed a faster recursive

programming technique to obtain the optimal threshold for digital image segmentation. Thus, (5) can be expressed as (6).

$$f_{MCET}(th) = \left(\sum_{i=1}^{L} ih(i)log(i) - \sum_{i=1}^{th-1} ih(i)log(\mu(1, th)) - \sum_{i=th}^{L} ih(i)log(\mu(th, L+1)) \right) \quad (6)$$

The bilevel approach in (6) is extended to make the multilevel approach using the vector **th**, which contains nt threshold values, shown in (7).

$$f_{MCET}(\mathbf{th}) = \sum_{i=1}^{L} ih(i)log(i) - \sum_{i=1}^{nt} H_i \quad (7)$$

where nt is the total number of thresholds and H_i is determined as (8).

$$\begin{aligned} H_1 &= \sum_{i=1}^{th_1-1} ih(i)log(\mu(1, th_1)) \\ H_k &= \sum_{i=th_{k-1}}^{th_k-1} ih(i)log(\mu(th_{k-1}, th_k)), \quad 1 < k < nt \\ H_{nt} &= \sum_{i=th_{nt}}^{L} ih(i)log(\mu(th_{nt}, L+1)) \end{aligned} \quad (8)$$

The process of minimizing the statistical cross-entropy criterion to determine the optimal threshold is carried out within the proposed HH algorithm. In this context, the objective function $f_{MCET}(\mathbf{th})$ in (7) iteratively guides the algorithm to explore and refine the threshold values, thus improving the quality of image segmentation.

In addressing the challenge of multilevel image segmentation, the paper employs the MCET criterion as a pivotal segmentation tool. This choice is based on a detailed mathematical model of the problem, where the task of pixel classification is transformed into a well-defined optimization problem. The formulation is designed to capture the intricacies and complexities of multilevel image segmentation.

The mathematical foundation provided by MCET facilitates the translation of the problem into an optimization objective function, seeking to minimize the statistical cross-entropy criterion. As detailed before, this criterion operates as a robust measure of the divergence between the original grayscale image and its segmented counterpart. The approach models the mathematical complexities inherent in multilevel image segmentation to identify optimal threshold values that not only meet the stringent requirements of the task but also contribute to solving practical engineering problems in image processing.

IV. HYPER-HEURISTIC GENETIC ALGORITHM BASED ON THOMPSON SAMPLING WITH DIVERSITY

In this section, we present the proposed approach, which is called Hyper-heuristic Genetic Algorithm based on Thompson Sampling with Diversity (HHGATSD). HHGATSD incorporates principles inspired by the HH based on Thompson Sampling (HHTS) for the dynamic selection of crossover and mutation operators [61]. It is important to note that HHGATSD stands out as a distinctive algorithm by integrating this operator selection strategy into a comprehensive GA framework, thereby enhancing its performance in addressing optimization problems.

HHGATSD introduces two pivotal stages, namely a diversity stage and a cloning stage, strategically placed to address the challenge of balancing the exploration and exploitation of the algorithm. The diversity stage employs the Normalized True Diversity (D_{TD}) metric, offering a dynamic measure of population diversity during the evolutionary process. Additionally, the cloning stage, guided by a carefully tuned α parameter, improves exploration by generating clones of the worst-performing individuals. Collectively, these innovative components contribute to the robustness, scalability, and efficiency of the algorithm in solving complex optimization problems.

The flowchart in Fig. 1 visually illustrates the structural similarities between HHGATSD and conventional GA. It integrates additional stages to augment performance and the ability to find high-quality solutions, with MAB theory blocks highlighted using a rectangle with dotted lines.

The main steps of the HHGATSD are presented below:

1) Initialization: The process begins by defining the parameters and randomly generating uniformly distributed particles as the initial population. Furthermore, the diversity of the distribution of the particles in the population is obtained and quantified as the value of the normalization method known as Normalization with Maximum Diversity so Far (NMDF) [62]. After the initialization phase, we obtain and evaluate each individual's fitness in the population using the objective function.
2) Fitness Evaluation: After generating the initial population, each candidate solution is evaluated using the objective function. The best solution is determined based on the smallest fitness value within the population, marking the beginning of the iterative process.
3) Parent Selection: This stage selects individuals from the current population to act as parents in the crossover and mutation process, producing offspring that may become part of the next generation. The Roulette Wheel Selection strategy is utilized, where each individual's probability of being selected is proportional to their fitness [63]. Individuals with lower fitness have a higher probability of being chosen as parents, but there is no guarantee that the best individuals will be selected.
4) Operator Selection: This is the first additional stage of the HH that is introduced. Instead of using fixed crossover and mutation operators, we treat the operator selection as a multi-arm bandit (MAB) problem [64]. Thompson sampling is employed since it outperforms other algorithms in tackling the MAB problem [45].

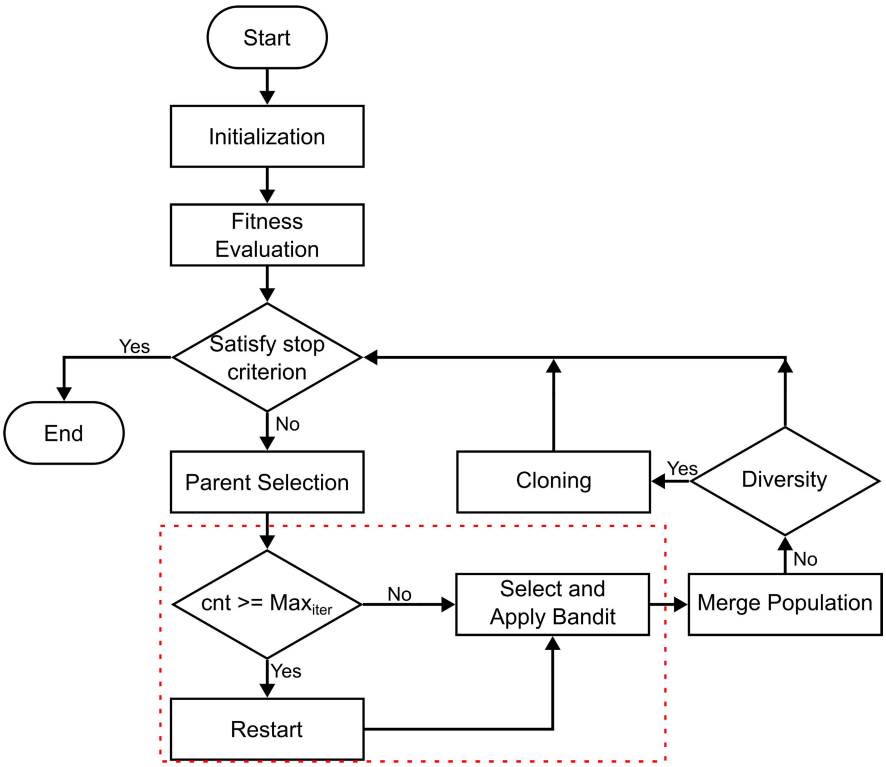


FIGURE 1. Flowchart of the proposed HHGATSD.

It dynamically selects the most promising operators through stochastic sampling based on estimated rewards, optimizing the operator combination for the highest cumulative reward over *Iter* iterations. The pseudocode of this process is presented in Algorithm 1, and the output is the action that contains the combination of heuristics that will be applied. The process of parent selection and operator selection is repeated N times.

Algorithm 1 Pseudocode of *TS*

Inputs: $\boldsymbol{R}$, $\boldsymbol{P}$, *cnt*
if ($cnt \leq Iter$) **then**
  Reinitialize $\boldsymbol{R}$, $\boldsymbol{P}$ and *cnt*
else
  $cnt \leftarrow cnt + 1$
end if
for (i $\leftarrow$ 1 to $num_{actions}$) **do**
  $\theta_i \leftarrow Beta(R_i{+}1, P_i{+}1)$ sample from Beta distribution
end for
acion $\leftarrow$ Select the action with the $argmax(\boldsymbol{\theta})$
Return *acion*

5) Merge Population: A set of N new individuals is generated after applying operators. Their fitness is evaluated using the objective function, and the generational replacement and sorting strategy selects the new population [65].

6) Diversity: In this stage, we calculate the D_{TD}, a metric for quantifying diversity within the population. This measure represents the average standard deviation of the position of each individual and is obtained using in (9) [66]. The diversity stage evaluates D_{TD} in each iteration and checks if it is less than 0.2. Additionally, the diversity stage verifies whether the number of accesses to the objective function is lower than half of Max_{Acc}. Meeting these conditions triggers the cloning stage.

$$D_{TD} = \frac{\frac{1}{N}\sqrt{\sum_{k=1}^{N}(\overline{x_k^2} - (\overline{x_k})^2)}}{NMDF} \tag{9}$$

where

$$\overline{x_k^2} = \frac{1}{D}\sum_{i=1}^{D} x_{i,k}^2 \tag{10}$$

such that $i = 1, 2, \ldots, D$, where D represents the dimension of the problem and $k = 1, 2, \ldots, N$, where N is the number of particles in the population.

7) Cloning: The cloning phase, described in (11), aims to generate clones of the $0.3N$ worst individuals in the population, evaluate their cost, and replace them in the population with the most promising clones. This phase contributes to increasing the diversity and exploration in the search space of the proposed HHs.

$$x_{i,k} = x_{i,k} + \xi \tag{11}$$

where ξ is a random number between -α and α, and α is a positive parameter that controls the search domain.

8) Stop Criterion: The algorithm evaluates if it has reached Max_{Acc} to the objective function allowed. If not, it returns to the Parent Selection step (Step 3), repeating the evolutionary cycle. This stop criterion is crucial for managing computational resources and preventing indefinite algorithm execution, ensuring completion within a reasonable time frame while controlling the total number of objective function accesses.

The set of crossover and mutation operators to be applied to the parents selected in Step 4 in the proposed HHGATSD algorithm is described below. This approach applies four crossover operators, which are:

- UNDX - unimodal normal distribution crossover [67]
- PCX - parent centric recombination [68]
- BLX - blend crossover [69]
- Single point crossover [70]

Four mutation operators described below are also considered:

- Michalewicz' mutation [71]
- DE mutation [72]
- Random mutation [73]
- OBL - opposition-based learning [74]

There are two crucial parameters in HHGATSD. The first parameter is the variable *Iter*, which is involved in Step 4 and represents the maximum cumulative reward across iterations. The role of this variable is to re-initialize the reward (**R**) and penalty (**P**) vectors, essential components for the TS algorithm. The significance of resetting this algorithm lies in its adaptability to the changing dynamics of the environment, ensuring a continuous and efficient search space exploration. Therefore, the variable *Iter* plays a determinant role.

The second parameter is α in Step 5, a positive parameter controlling the search in the cloning phase. This value determines the extent of the random adjustments applied to the solutions during the cloning phase.

V. EXPERIMENTAL FRAMEWORK

In this section, we discuss the datasets utilized in this research for evaluating the performance of the proposed algorithm. We also describe the metrics employed and the algorithms used for comparison.

A. DATASETS

We evaluated the proposed method on three datasets: (1) 29 functions from the well-known IEEE CEC2017 benchmark for solving constrained real parameter optimization problems [75], (2) ten random images from the Berkeley Segmentation Dataset (BSDS500) [76], and (3) a set of 612 frames from a traffic camera video.

First, we extensively verify and compare the efficiency of HHGATSD with other algorithms on the challenging IEEE CEC2017 benchmark for 30 and 50 dimensions, which exhibits high complexity. Table 1 presents the functions and features of this benchmark along with their respective optimal or minimum values. The test set contains 29 benchmark functions, excluding f_2 from the set, due to the unstable behavior [77]. It includes four sets of functions: unimodal functions (f_1), multimodal functions ($f_3 - f_9$), hybrid functions ($f_{10} - f_{19}$), and composition functions ($f_{20} - f_{30}$).

TABLE 1. Summary of the CEC2017 benchmark functions.

Type	No.	Functions	$F_i^* = F_i(x^*)$
Unimodal	1	Shifted and Rotated Bent Cigar Function	100
Multimodal	3	Shifted and Rotated Rosenbrock's Function	300
	4	Shifted and Rotated Rastrigin's Function	400
	5	Shifted and Rotated Expanded Scaffer's F6 Function	500
	6	Shifted and Rotated Lunacek Bi_Rastrigin Function	600
	7	Shifted and Rotated Non-Continuous Rastrigin's Function	700
	8	Shifted and Rotated Levy Function	800
	9	Shifted and Rotated Schwefel's Function	900
Hybrid	10	Hybrid Function 1 (N=3)	1000
	11	Hybrid Function 2 (N=3)	1100
	12	Hybrid Function 3 (N=3)	1200
	13	Hybrid Function 4 (N=4)	1300
	14	Hybrid Function 5 (N=4)	1400
	15	Hybrid Function 6 (N=4)	1500
	16	Hybrid Function 7 (N=5)	1600
	17	Hybrid Function 8 (N=5)	1700
	18	Hybrid Function 9 (N=5)	1800
	19	Hybrid Function 10 (N=6)	1900
Composition	20	Composition Function 1 (N=3)	2000
	21	Composition Function 2 (N=3)	2100
	22	Composition Function 3 (N=4)	2200
	23	Composition Function 4 (N=4)	2300
	24	Composition Function 5 (N=5)	2400
	25	Composition Function 6 (N=5)	2500
	26	Composition Function 7 (N=6)	2600
	27	Composition Function 8 (N=6)	2700
	28	Composition Function 9 (N=3)	2800
	29	Composition Function 10 (N=3)	2900
	30	Composition Function 11 (N=3)	3000
Search range: $[-100, 100]^D$			

Second, we used the UC Berkeley image segmentation benchmark, which is a dataset commonly referenced in the image processing literature. This dataset consists of 500 natural images that exhibit different levels of complexity in their intensities. We randomly selected ten images from this benchmark to create our second evaluation set for assessing the proposed method. In Fig. 2, we show the benchmark images utilized in this study, along with their corresponding histograms. These images can be accessed at the following link,[1] serve as a crucial component of our evaluation. They represent a diverse array of real-world scenarios, allowing us to assess the robustness and adaptability of the proposed method across different levels of image complexity.

Finally, we use a third set of images obtained from a stationary road traffic surveillance camera, which can be found at the following link.[2] We apply a multilevel segmentation to this set using the proposed method for subsequent post-processing to identify the vehicles on the road. In Fig. 3, we present eight randomly selected frames out of a total of 612 as sample images representing the third set.

B. EXPERIMENTAL SETUP

The performance of various MH optimizers in the first and second datasets was evaluated. Specifically, GA, PSO, DE,

[1] www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html

[2] www.kaggle.com/foyecey/traffic-road

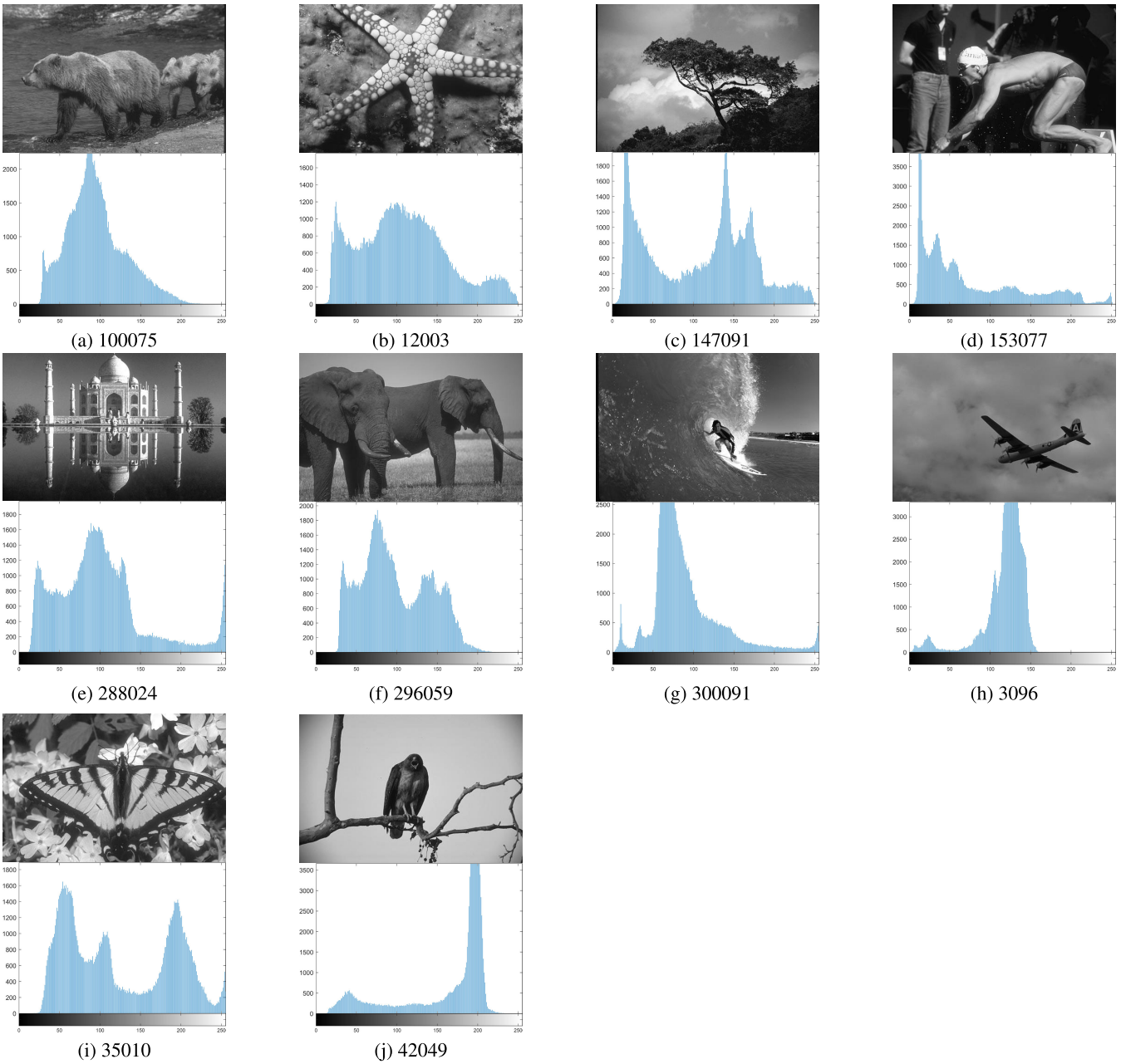


FIGURE 2. Set of benchmark images with their respective histogram. The histograms offer insights into the frequency of intensity values in each image, providing additional information about the image features.

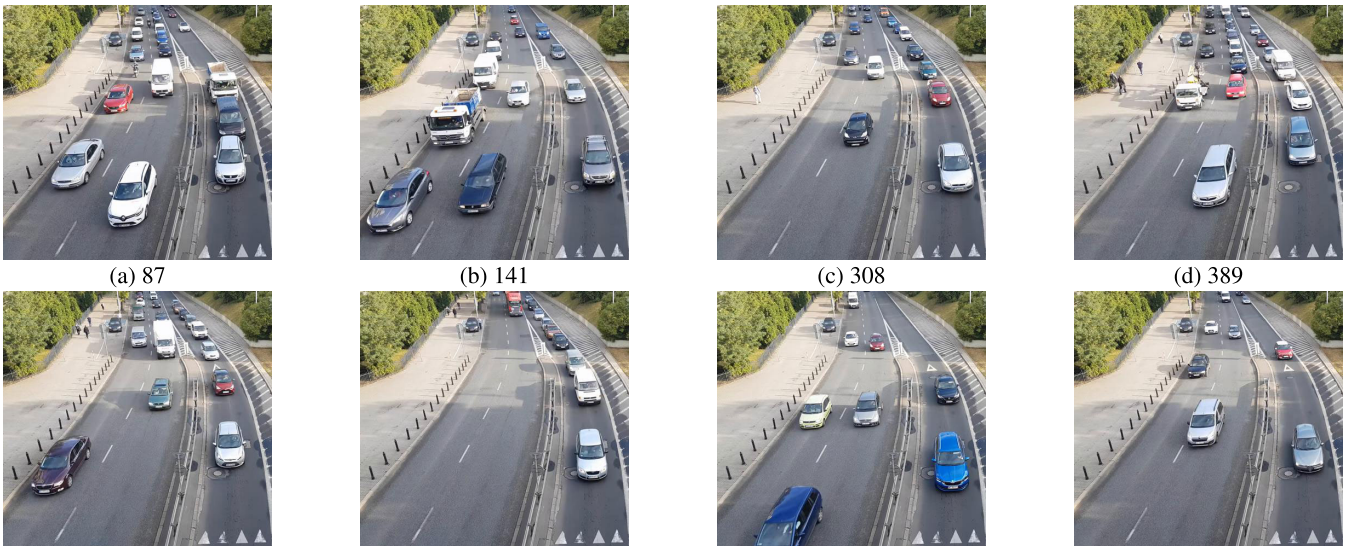
BSD, GWO, HHO, JADE, and BLPSO were considered. The internal parameters of these algorithms were configured based on their original references to ensure their optimal performance.

As part of our analysis of the third dataset, we meticulously assessed our proposed algorithm's efficacy compared to several established segmentation methods. Specifically, our evaluation encompassed a comparison with the widely recognized Otsu's method [78], the classical K-means algorithm [79], and the Fuzzy IterAg algorithm [80].

Across all experiments conducted on the IEEE CEC2017 benchmark in 30 and 50 dimensions (D), the stop criteria established for all the algorithms are 50,000 and 80,000 function accesses, respectively.

Experiments for the Berkeley image benchmark were conducted using 2, 4, 6, 8, and 10 thresholds for the multilevel thresholding problem. The objective function employed was the MCET approach, and a maximum of 100 iterations were allowed for each experiment as the stop criterion. This criterion guided the search for optimal thresholds, ensuring effective pixel classification in the resulting segmented images.

The total number of search agents in the population, denoted as N, was consistently set at 50 for all test algorithms.



(a) 87 (b) 141 (c) 308 (d) 389

(e) 416 (f) 441 (g) 507 (h) 534

FIGURE 3. Examples of frames from the road traffic surveillance camera.

Furthermore, due to the stochastic nature of MAs, each method underwent 30 independent runs (T_E) to establish its robustness. It is noteworthy that, for the proposed algorithm, the internal parameters *Iter* and α were set to 200 and 5, respectively. All experiments were conducted using Matlab version 9.8 on hardware equipped with a 1.60GHz Intel Core i5 CPU with 16GB of RAM.

C. METHODOLOGY FOR VEHICLE SEGMENTATION

In order to effectively identify and segment vehicles within the images acquired from the traffic surveillance camera and to validate the proposed algorithm in a real-world environment, we employ a comprehensive methodology that is concisely described in the flowchart presented in Figure 4. This figure shows the output of each of the most crucial steps.

Each step plays a key role in achieving the objective. The methodology initiates the process with Robust Principal Component Analysis (RobustPCA) as the initial data processing step. When applied to a matrix comprising the 612 frames extracted from the camera video, RobustPCA effectively decomposes the information into two primary components: a low-rank matrix (L) representing underlying structural information and a high-dimensional sparse matrix (S) capturing noise and unwanted variations. Matrix L is associated with stationary or background elements in the images, while matrix S detects moving vehicles and dynamic objects. The resulting matrix of S is denoted as $\mathbf{I_S}$ and becomes the basis for the subsequent analysis.

It is important to note that the RobustPCA technique relies on two key parameters: λ and max_{iter}. The parameter λ provides a normalized value based on the size of the input matrix $\mathbf{X}$. The choice of $\lambda/10$ is set empirically to balance the algorithm's robustness against outliers while ensuring efficient decomposition. Additionally, max_{iter}, representing the maximum number of iterations, is set to 1000 to ensure convergence while controlling computational resources. This value is selected through empirical experimentation, seeking a balance between achieving convergence and avoiding excessive computation time.

The next step involves multilevel segmentation, achieved by applying five thresholds to divide $\mathbf{I_S}$ into regions corresponding to specific vehicle attributes. The outcome of this multilevel segmentation process is named $\mathbf{I_{th}}$, which is subsequently binarized to generate a mask, $\mathbf{I_B}$, considering only the highest threshold.

In the post-processing phase, morphological operations are applied to refine the $\mathbf{I_B}$ mask in preparation for further analysis. This step involves erosion and morphological closure to enhance region cohesion. Erosion employs a 2×2 pixel square-shaped structuring element to reduce noise and small artifacts. Conversely, morphological closure utilizes a 15-pixel radius disk as a structuring element to merge closely spaced regions, ensuring the integrity of vehicle shapes. The final binary mask, $\mathbf{I_{seg}}$, forms the solid basis for vehicle feature extraction and tracking algorithms, contributing to traffic analysis and management accuracy.

For a comprehensive overview of this methodology, the pseudocode is presented in the Algorithm 2.

D. METRICS

Since the algorithm presented in this paper is stochastic, the standard deviation (STD) described in (12) is used to test the stability of the optimizer. According to Ghamisi et al., if the STD value increases, the algorithm becomes more unstable [81].

$$STD = \sqrt{\sum_{i=1}^{T_E} \frac{\sigma_i - \mu}{T_E}} \tag{12}$$

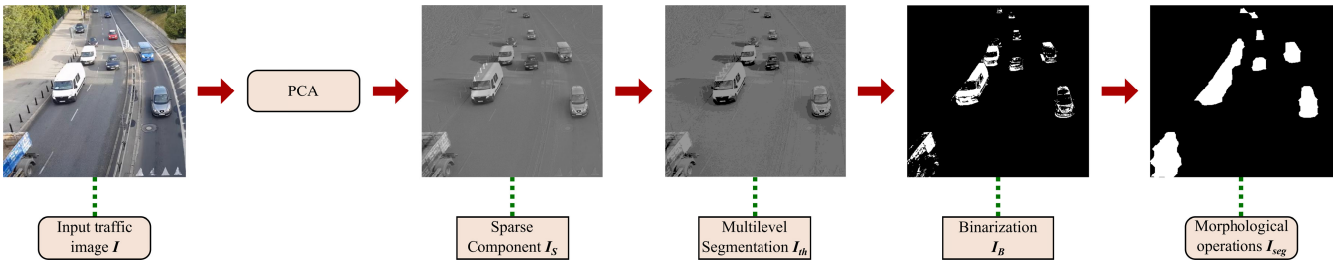


FIGURE 4. Flowchart of the proposed method for vehicle segmentation in a traffic surveillance camera.

Algorithm 2 Pseudocode of Vehicle Segmentation Methodology

Inputs: X
$\lambda \leftarrow 1/\sqrt{\max(\text{size}(\mathbf{X}))}$
$max_{iter} \leftarrow 1000$
$nt \leftarrow 5$
$[\mathbf{L}, \mathbf{S}] \leftarrow \text{RobustPCA}(\mathbf{X}, \lambda/10, max_{iter})$
$i \leftarrow$ index of the image to segment
$\mathbf{I_S} \leftarrow \mathbf{S_i}$
$\mathbf{th} \leftarrow \text{MultilevelMethod}(\mathbf{I_S}, nt)$
$\mathbf{I_{th}} \leftarrow \text{MultilevelSegmentation}(\mathbf{I_S}, \mathbf{th})$
for (i $\leftarrow$ 1 to num_{col}) **do**
  for (j $\leftarrow$ 1 to num_{rows}) **do**
    if $Ith(i,j) \geq th_{nt}$ **then**
      $I_B(i,j) \leftarrow 1$
    else
      $I_B(j,k) \leftarrow 0$
    end if
  end for
end for
$\mathbf{I_{seg}} \leftarrow \text{MorphologicalOperations}(\mathbf{I_B})$
Return $\mathbf{I_{seg}}$

To evaluate the multilevel segmentation quality performance of the proposed approach on the BSD500, the following three metrics are considered:

- Peak Signal to Noise Ratio (*PSNR*) [82]: *PSNR* compares the segmented and the original images using the Root Mean Square Error (*RMSE*) of each pixel. A higher PSNR value indicates that there is more similarity between the images. The mathematical formula for this metric is shown in equation (13).

$$PSNR = 20\log_{10}\frac{255}{RMSE}$$
$$RMSE = \sqrt{\frac{\sum_{i=1}^{ro}\sum_{j=1}^{co}[I_0^C(i,j) - I_{th}^C(i,j)]}{ro \times co}} \quad (13)$$

- Structural Similarity Index (*SSIM*) [83]: It is another measure of the similarity between original and segmented images used to identify the internal structures and is defined in (14). Better segmentation performances produce a higher *SSIM* value.

$$SSIM = \frac{(2\mu_{I_{Gr}}\mu_{I_{th}} + C_1)(2\sigma_{I_{Gr}I_{th}} + C_2)}{(\mu_{I_{Gr}}^2 + \mu_{I_{th}}^2 + C_1)(\sigma_{I_{Gr}}^2 + \sigma_{I_{th}}^2 + C_2)} \quad (14)$$

where $\mu_{I_{Gr}}$ and $\mu_{I_{th}}$ are the mean values of the original and the segmented image, respectively. For each image the values of $\sigma_{I_{Gr}}$ and $\sigma_{I_{th}}$ correspond to the standard deviation and $\sigma_{\mathbf{I_{Gr}I_s}}$ is the covariance of I_{Gr} and I_{th}. C_1 and C_2 are two variables for weak denominator stabilization and the values used are $C_1 = C_2 = 0.065$.

- Feature Similarity Index (*FSIM*) [84]: *FSIM* is a metric that measures the similarity between both images regarding their internal features such as edges, corners, etc. It is mathematically defined in (15), where a higher value is interpreted as a better segmentation.

$$FSIM = \frac{\sum_{w\epsilon\Omega} S_L(w)PC_m(w)}{\sum_{w\epsilon\Omega} PC_m(w)} \quad (15)$$

where Ω represents the spatial domain of the image. S_L is the similarity that exists in the components. *PC* is the phase congruence, a dimensionless measure of the importance of local structure.

On the other hand, to evaluate the quality and performance of vehicle segmentation within traffic camera frames, we consider a comprehensive set of five metrics to compare the segmented and ground truth masks, shown below:

- Dice Coefficient (*DSC*) [85]: *DSC*, commonly known as F1-score, quantifies the extent of overlap between predicted and ground truth segmentation masks. A *DSC* value of 1 indicates a perfect overlap, while lower values suggest less agreement. It is particularly useful for measuring segmentation accuracy regarding both false positives and false negatives.
  The mathematical expression is outlined in (16). Here, A represents the set of pixels in the predicted mask, B represents the set of pixels in the ground truth mask, and $|\cdot|$ denotes the cardinality (number of elements) of a set.

$$DSC = \frac{2|A \cap B|}{|A| + |B|} \quad (16)$$

- Jaccard Index [86] (*IoU*): The Jaccard index, also known as Intersection over Union (*IoU*), provides a measure of similarity between predicted and ground truth masks. It is calculated by dividing the intersection of the two

sets by their union. A higher IoU value signifies better alignment, making it a valuable metric for assessing the accuracy of segmentation masks. This metric is shown in (17).

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{17}$$

- Precision (P) [87]: P measures the accuracy of positive predictions in the generated mask. It is particularly crucial to evaluate how well the segmentation model identifies true positive (TP) instances while minimizing false positives (FP). A higher precision value indicates a more accurate identification of positive pixels. Mathematically, P is defined in (18), TP represents the count of positive pixels accurately identified, and FP signifies the number of pixels erroneously classified as positive.

$$P = \frac{TP}{TP + FP} \tag{18}$$

- Recall (R) [88]: R, also known as sensitivity, measures the ability of the algorithm to identify all positive instances in the mask. It is a crucial metric for minimizing false negatives, indicating the model's sensitivity to detecting actual positive instances in the segmentation. In (19) formulates the R metric, where FN (False Negatives) represents the count of positive pixels misclassified as negative.

$$R = \frac{TP}{TP + FN} \tag{19}$$

- Accuracy (ACC) [88]: ACC provides an overall measure of correctness and is not specific to the positive class. It considers both true positive and true negative instances in the segmentation mask, making it a comprehensive metric for evaluating the overall performance of the segmentation model. The mathematical formula for this metric is shown in 20, and TN (True Negatives) is the number of correctly identified negative pixels.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{20}$$

E. NON-PARAMETRIC TEST

Due to the intrinsic nature of metaheuristic algorithms, a robust and statistically rigorous comparison and the validation of numerical results require the use of non-parametric statistical tests. These tests prove particularly appropriate when the data fails to meet the assumptions of normality and does not rely on specific probability distributions. Consequently, they are well-suited for assessing the effectiveness of our proposed method and discerning its significance relative to other algorithms [89].

In this work, we employ the Friedman test, which allows us to evaluate whether there are statistically significant differences in the performance of the algorithms on the benchmark functions. To enhance the precision of our analysis, we implement Holm's p-correction value method [90]. This method is vital for controlling the family error rate when conducting multiple pairwise comparisons, ensuring that random chance does not influence our results.

By incorporating these robust statistical techniques into our evaluation, we aim to provide a comprehensive and reliable assessment of the HHGATSD algorithm's effectiveness compared to other methods. This approach enables us to confidently identify which algorithms exhibit superior performance across the benchmark function set.

VI. RESULT

This section provides results from three experimental trials to conduct a comprehensive analysis of the performance of HHGATSD in different scenarios. The objective is to evaluate the effectiveness of the proposed algorithm in tackling complex high-dimensional optimization problems and to gauge its adaptability and robustness for real-world problems such as multilevel image segmentation.

A. EXPERIMENTS WITH IEEE CEC2017 BENCHMARK

In Tables 2 and 3, we present our experiments' comprehensive results with the IEEE CEC2017 benchmark functions in 30 and 50 dimensions, respectively. Each row corresponds to a specific benchmark function, and each column represents an optimization algorithm used in this comparison.

The tables provide two key statistics for each algorithm in each benchmark function: AVG and STD of the fitness. The algorithm achieving the best AVG result with the lowest fitness in each benchmark function is denoted in blue, and the results with the lowest STD values are highlighted in bold, making them easy to identify. Additionally, the 'W/T/L' row at the bottom quantifies the number of wins (W), ties (T), and losses (L) for each algorithm in comparison to others across all benchmark functions.

In the context of the high-dimensional optimization problems of the IEEE CEC2017 benchmark functions in 30 dimensions, the HHGATSD algorithm demonstrates its skills. By analyzing the results presented in Table 2, it is evident that the algorithm consistently outperforms the other algorithms used in this comparison in average fitness values with significantly reduced dispersion, as indicated by low standard deviation values for most of the functions.

When examining the 'W/T/L' row, HHGATSD stands out with 18 wins, 0 ties, and 11 losses in average fitness. However, a comparison based on standard deviation metrics obtained 8 wins, 0 ties, and 21 losses, a performance slightly below that of the BSDE algorithm.

On the other hand, in 50 dimensions, the HHGATSD algorithm obtained better results. As shown in Table 3, it maintained and increased performance, outperforming the other algorithms in average and standard deviation fitness values.

Examining the 'W/T/L' row, HHGATSD again excelled by obtaining 19 wins, 0 draws, and 11 losses in average fitness. Similarly, it stood out in the standard deviation averages comparison with 13 wins, 0 ties, and 16 losses, which

TABLE 2. Statistical results of the CEC2017 benchmark functions in 30 dimensions.

Instance		HHGATSD	GA	PSO	DE	JADE	BSDE	GWO	HHO
F1	AVG	5.48E+03	1.27E+05	**2.31E+03**	1.70E+09	5.03E+03	2.48E+03	1.62E+09	2.84E+07
	STD	1.24E+04	8.44E+04	2.84E+03	1.14E+09	6.06E+03	**1.28E+03**	1.36E+09	6.35E+06
F3	AVG	1.32E+04	1.28E+05	**5.72E+03**	5.19E+04	1.89E+05	6.44E+04	4.73E+04	3.29E+04
	STD	5.00E+03	3.57E+04	**2.84E+03**	1.46E+04	3.07E+04	1.16E+04	1.11E+04	7.00E+03
F4	AVG	5.06E+02	5.12E+02	5.12E+02	6.81E+02	**4.86E+02**	5.10E+02	5.88E+02	5.67E+02
	STD	3.53E+01	2.91E+01	6.13E+01	1.32E+02	1.41E+01	**1.11E+01**	6.41E+01	4.53E+01
F5	AVG	**5.35E+02**	5.82E+02	6.48E+02	5.39E+02	5.94E+02	5.85E+02	6.02E+02	7.55E+02
	STD	1.05E+01	2.26E+01	3.45E+01	1.18E+01	1.59E+01	**1.02E+01**	2.21E+01	3.86E+01
F6	AVG	**6.00E+02**	6.01E+02	6.28E+02	6.01E+02	6.03E+02	6.01E+02	6.07E+02	6.64E+02
	STD	**2.34E-02**	2.82E-02	1.40E+01	9.49E-01	1.29E+00	2.90E-01	2.95E+00	5.68E+00
F7	AVG	8.20E+02	8.09E+02	8.01E+02	**8.00E+02**	8.21E+02	8.20E+02	8.76E+02	1.26E+03
	STD	3.06E+01	2.19E+01	2.37E+01	3.11E+01	1.59E+01	**8.56E+00**	5.38E+01	6.58E+01
F8	AVG	**8.37E+02**	8.83E+02	9.09E+02	8.43E+02	8.92E+02	8.83E+02	8.91E+02	9.67E+02
	STD	1.35E+01	2.42E+01	2.16E+01	1.56E+01	1.60E+01	**1.12E+01**	3.32E+01	2.62E+01
F9	AVG	**9.10E+02**	1.29E+03	2.86E+03	1.06E+03	2.80E+03	1.70E+03	1.70E+03	8.15E+03
	STD	**1.68E+00**	3.44E+02	1.29E+03	1.02E+02	1.18E+03	3.50E+02	6.79E+02	8.53E+02
F10	AVG	**3.17E+03**	4.72E+03	4.37E+03	7.52E+03	5.15E+03	4.62E+03	4.92E+03	5.88E+03
	STD	6.71E+02	6.95E+02	6.75E+02	1.03E+03	**3.07E+02**	4.14E+02	1.45E+03	7.57E+02
F11	AVG	**1.21E+03**	2.48E+03	1.22E+03	1.51E+03	3.76E+03	1.23E+03	1.65E+03	1.29E+03
	STD	4.70E+01	1.32E+03	3.32E+01	3.90E+02	1.26E+03	**2.19E+01**	4.57E+02	4.30E+01
F12	AVG	2.43E+07	3.65E+06	**6.82E+05**	5.62E+07	2.95E+06	7.20E+05	6.32E+07	1.93E+07
	STD	1.69E+07	2.59E+06	4.67E+05	7.60E+07	4.05E+06	**3.67E+05**	6.86E+07	9.66E+06
F13	AVG	**2.79E+03**	4.70E+04	1.22E+04	1.93E+07	2.41E+06	1.25E+04	4.87E+06	7.49E+05
	STD	**1.47E+03**	3.71E+04	1.02E+04	3.98E+07	2.89E+06	4.93E+03	1.68E+07	9.00E+05
F14	AVG	**1.56E+03**	9.54E+05	1.81E+04	1.61E+05	8.58E+05	4.51E+04	2.73E+05	3.82E+05
	STD	**3.29E+02**	9.01E+05	1.70E+04	3.23E+05	9.05E+05	3.69E+04	3.81E+05	5.02E+05
F15	AVG	5.09E+03	1.37E+04	7.64E+03	3.78E+05	1.48E+06	**2.20E+03**	3.56E+05	9.55E+04
	STD	1.96E+03	2.07E+04	7.38E+03	1.14E+06	1.92E+06	**6.73E+02**	9.29E+05	6.07E+04
F16	AVG	**2.06E+03**	2.61E+03	2.54E+03	2.10E+03	2.71E+03	2.31E+03	2.47E+03	3.44E+03
	STD	2.68E+02	2.94E+02	2.75E+02	2.46E+02	2.11E+02	**1.30E+02**	3.50E+02	4.22E+02
F17	AVG	1.88E+03	2.22E+03	2.20E+03	1.83E+03	2.23E+03	**1.87E+03**	2.01E+03	2.60E+03
	STD	7.56E+01	1.97E+02	2.09E+02	9.82E+01	1.52E+02	**7.52E+01**	1.58E+02	3.36E+02
F18	AVG	**1.23E+05**	2.46E+06	2.77E+05	6.35E+05	2.60E+06	1.83E+05	1.17E+06	1.92E+06
	STD	1.54E+05	2.38E+06	2.70E+05	1.13E+06	1.34E+06	**9.71E+04**	1.83E+06	2.42E+06
F19	AVG	**3.32E+03**	1.36E+04	1.02E+04	2.02E+05	7.38E+05	4.58E+03	4.11E+06	6.99E+05
	STD	**2.28E+03**	1.98E+04	9.31E+03	7.22E+05	6.85E+05	2.36E+03	1.06E+07	5.80E+05
F20	AVG	**2.14E+03**	2.55E+03	2.46E+03	2.14E+03	2.56E+03	2.24E+03	2.40E+03	2.76E+03
	STD	**7.12E+01**	2.18E+02	1.35E+02	8.97E+01	1.29E+02	8.95E+01	1.40E+02	2.27E+02
F21	AVG	**2.21E+03**	2.38E+03	2.43E+03	2.35E+03	2.41E+03	2.38E+03	2.39E+03	2.57E+03
	STD	**1.99E-03**	2.18E+01	2.65E+01	1.28E+01	2.02E+01	1.02E+01	3.80E+01	4.71E+01
F22	AVG	**2.28E+03**	4.11E+03	3.46E+03	3.45E+03	5.60E+03	2.33E+03	5.01E+03	6.79E+03
	STD	**2.19E-03**	2.13E+03	1.76E+03	1.99E+03	1.74E+03	3.06E+01	1.71E+03	1.50E+03
F23	AVG	2.84E+03	2.76E+03	2.90E+03	**2.72E+03**	2.76E+03	2.75E+03	2.77E+03	3.21E+03
	STD	1.40E+01	3.13E+01	6.54E+01	2.27E+01	2.27E+01	**1.07E+01**	5.14E+01	1.29E+02
F24	AVG	**2.60E+03**	2.93E+03	3.06E+03	2.89E+03	2.94E+03	2.92E+03	2.94E+03	3.44E+03
	STD	**0.00E+00**	3.42E+01	7.64E+01	2.71E+01	2.45E+01	1.87E+01	6.15E+01	1.38E+02
F25	AVG	2.94E+03	2.90E+03	2.90E+03	3.02E+03	**2.89E+03**	2.90E+03	3.00E+03	2.95E+03
	STD	1.30E+02	1.77E+01	1.39E+01	9.57E+01	**1.01E+01**	1.23E+01	3.84E+01	3.55E+01
F26	AVG	**3.18E+03**	4.86E+03	4.70E+03	4.43E+03	4.63E+03	4.03E+03	4.83E+03	7.32E+03
	STD	9.29E+02	3.64E+02	1.58E+03	**2.28E+02**	2.83E+02	8.30E+02	4.81E+02	1.27E+03
F27	AVG	3.66E+03	3.24E+03	3.27E+03	3.24E+03	**3.22E+03**	3.26E+03	3.26E+03	3.47E+03
	STD	9.28E+01	1.44E+01	4.13E+01	1.96E+01	1.01E+01	**7.02E+00**	2.87E+01	1.10E+02
F28	AVG	**3.24E+03**	3.25E+03	3.29E+03	3.68E+03	3.27E+03	3.28E+03	3.42E+03	3.32E+03
	STD	**1.41E+01**	2.92E+01	3.20E+01	3.15E+02	4.87E+01	1.84E+01	1.27E+02	3.87E+01
F29	AVG	**3.33E+03**	3.83E+03	3.95E+03	3.56E+03	3.79E+03	3.63E+03	3.82E+03	4.81E+03
	STD	1.58E+02	2.14E+02	1.97E+02	1.40E+02	1.20E+02	**7.09E+01**	1.80E+02	4.74E+02
F30	AVG	2.76E+04	1.77E+04	**1.58E+04**	2.58E+05	3.57E+05	2.75E+04	6.35E+06	4.81E+06
	STD	1.88E+04	**6.58E+03**	1.06E+04	3.70E+05	3.34E+05	1.52E+04	5.08E+06	3.96E+06
W/T/L	AVG	**18/0/11**	0/0/29	4/0/25	2/0/27	3/0/26	2/0/27	0/0/29	0/0/29
	STD	8/0/21	1/0/28	1/0/28	1/0/28	2/0/27	**13/0/16**	0/0/0	0/0/0

tied with the BSDE algorithm. In particular, HHGATSD outperforms other methods even on complex and high-dimensional problems, highlighting its adaptability and robustness.

Table 4 presents the results obtained by applying non-parametric tests, specifically the Friedman and Holm tests, to evaluate the performance of the algorithms on the IEEE CEC2017 benchmark functions under two-dimensional

TABLE 3. Statistical results of the CEC2017 benchmark functions in 50 dimensions.

Instance		HHGATSD	GA	PSO	DE	JADE	BSDE	GWO	HHO
F1	AVG	2.24E+04	4.98E+05	3.97E+08	7.92E+09	**4.99E+03**	6.21E+03	7.56E+09	9.88E+07
	STD	2.52E+04	2.16E+05	7.20E+08	3.33E+09	5.78E+03	**3.71E+03**	3.31E+09	1.90E+07
F3	AVG	**2.94E+04**	3.01E+05	4.77E+04	1.95E+05	3.51E+05	1.58E+05	9.82E+04	8.75E+04
	STD	**7.67E+03**	5.38E+04	1.28E+04	3.72E+04	4.78E+04	1.94E+04	1.37E+04	1.77E+04
F4	AVG	5.54E+02	5.68E+02	5.81E+02	1.39E+03	**4.89E+02**	5.66E+02	1.11E+03	7.10E+02
	STD	4.53E+01	4.85E+01	7.38E+01	5.56E+02	5.48E+01	**3.82E+01**	4.69E+02	7.02E+01
F5	AVG	**5.92E+02**	6.81E+02	7.45E+02	6.08E+02	6.76E+02	7.00E+02	7.25E+02	8.95E+02
	STD	**1.80E+01**	4.83E+01	3.33E+01	2.20E+01	2.26E+01	2.14E+01	5.80E+01	3.52E+01
F6	AVG	**6.00E+02**	6.01E+02	6.43E+02	6.03E+02	6.03E+02	6.01E+02	6.17E+02	6.74E+02
	STD	**2.70E-02**	3.73E-02	8.06E+00	1.48E+00	1.95E+00	5.20E-01	4.62E+00	5.04E+00
F7	AVG	9.17E+02	9.30E+02	**9.03E+02**	9.76E+02	9.38E+02	9.54E+02	1.08E+03	1.82E+03
	STD	5.17E+01	4.19E+01	3.90E+01	8.13E+01	**2.40E+01**	2.55E+01	8.15E+01	9.55E+01
F8	AVG	**8.99E+02**	9.69E+02	1.07E+03	9.11E+02	9.91E+02	9.95E+02	1.03E+03	1.19E+03
	STD	2.54E+01	2.53E+01	3.65E+01	**1.57E+01**	2.68E+01	2.30E+01	6.37E+01	3.92E+01
F9	AVG	**9.86E+02**	4.81E+03	1.32E+04	2.05E+03	5.39E+03	7.33E+03	6.84E+03	2.62E+04
	STD	**1.02E+02**	3.89E+03	5.35E+03	9.73E+02	3.71E+03	1.33E+03	3.00E+03	2.99E+03
F10	AVG	**5.81E+03**	7.49E+03	6.81E+03	1.39E+04	8.21E+03	7.96E+03	7.09E+03	9.11E+03
	STD	9.20E+02	8.02E+02	7.86E+02	5.26E+02	3.80E+02	**3.06E+02**	9.08E+02	8.03E+02
F11	AVG	1.58E+03	6.72E+03	**1.26E+03**	4.22E+03	2.10E+04	1.41E+03	4.10E+03	1.53E+03
	STD	4.15E+02	5.21E+03	**4.02E+01**	2.00E+03	8.14E+03	4.27E+01	1.81E+03	9.87E+01
F12	AVG	**1.28E+05**	9.19E+06	6.55E+07	1.66E+09	3.55E+06	2.47E+06	7.86E+08	1.00E+08
	STD	**1.21E+05**	4.17E+06	2.18E+08	2.42E+09	4.93E+06	9.38E+05	9.03E+08	3.59E+07
F13	AVG	2.48E+04	3.59E+05	2.30E+06	4.67E+08	1.61E+06	**2.81E+03**	2.14E+08	3.96E+06
	STD	1.70E+04	7.72E+05	7.65E+06	1.15E+09	3.91E+06	**1.31E+03**	2.43E+08	3.49E+06
F14	AVG	**6.97E+03**	2.83E+06	1.16E+05	1.11E+06	3.42E+06	2.97E+05	7.12E+05	1.57E+06
	STD	**4.44E+03**	2.46E+06	9.92E+04	9.49E+05	2.01E+06	1.75E+05	5.60E+05	1.63E+06
F15	AVG	**3.32E+03**	1.17E+05	1.07E+04	1.59E+07	1.43E+06	8.72E+03	9.83E+06	4.84E+05
	STD	**1.39E+03**	1.29E+05	7.98E+03	4.18E+07	1.93E+06	2.97E+03	2.07E+07	2.04E+05
F16	AVG	2.75E+03	3.51E+03	3.00E+03	**2.65E+03**	3.65E+03	2.67E+03	3.11E+03	4.45E+03
	STD	2.78E+02	2.98E+02	5.50E+02	4.42E+02	2.57E+02	**1.74E+02**	5.22E+02	5.31E+02
F17	AVG	2.45E+03	3.17E+03	2.80E+03	**2.43E+03**	3.33E+03	2.63E+03	2.88E+03	3.84E+03
	STD	2.71E+02	3.03E+02	2.47E+02	2.76E+02	1.74E+02	**9.68E+01**	3.48E+02	3.91E+02
F18	AVG	**7.63E+05**	6.64E+06	1.05E+06	1.65E+06	1.13E+07	1.32E+06	5.47E+06	4.61E+06
	STD	**5.49E+05**	6.96E+06	8.02E+05	1.60E+06	8.35E+06	6.18E+05	1.01E+07	4.53E+06
F19	AVG	**5.37E+03**	1.92E+04	1.53E+04	8.66E+06	2.29E+05	1.83E+04	4.58E+06	1.28E+06
	STD	**3.36E+03**	1.17E+04	1.04E+04	2.68E+07	2.65E+05	4.30E+03	9.70E+06	1.20E+06
F20	AVG	**2.41E+03**	3.23E+03	2.93E+03	2.46E+03	3.42E+03	2.66E+03	2.88E+03	3.40E+03
	STD	1.95E+02	3.31E+02	2.88E+02	2.77E+02	1.71E+02	**1.56E+02**	2.71E+02	2.73E+02
F21	AVG	**2.29E+03**	2.47E+03	2.57E+03	2.41E+03	2.50E+03	2.49E+03	2.51E+03	2.87E+03
	STD	3.84E+01	3.83E+01	5.02E+01	2.13E+01	3.51E+01	**1.35E+01**	4.92E+01	8.66E+01
F22	AVG	**2.30E+03**	9.32E+03	8.64E+03	1.40E+04	9.92E+03	9.33E+03	8.88E+03	1.12E+04
	STD	**2.16E+01**	1.52E+03	1.61E+03	3.13E+03	4.26E+02	2.02E+03	1.96E+03	9.93E+02
F23	AVG	3.15E+03	2.96E+03	3.21E+03	**2.90E+03**	2.95E+03	2.97E+03	2.97E+03	3.82E+03
	STD	3.99E+01	4.96E+01	1.09E+02	4.34E+01	3.86E+01	**2.04E+01**	5.95E+01	1.65E+02
F24	AVG	**2.60E+03**	3.15E+03	3.40E+03	3.09E+03	3.14E+03	3.16E+03	3.12E+03	4.20E+03
	STD	**0.00E+00**	6.14E+01	1.01E+02	5.79E+01	4.85E+01	2.57E+01	7.14E+01	2.00E+02
F25	AVG	**3.03E+03**	3.08E+03	3.06E+03	3.89E+03	3.04E+03	3.12E+03	3.63E+03	3.22E+03
	STD	3.07E+01	2.74E+01	2.97E+01	4.90E+02	3.74E+01	**2.11E+01**	4.05E+02	5.64E+01
F26	AVG	**2.80E+03**	6.21E+03	5.95E+03	5.67E+03	5.74E+03	6.47E+03	6.48E+03	1.09E+04
	STD	**0.00E+00**	5.15E+02	2.39E+03	4.96E+02	3.09E+02	4.55E+02	6.11E+02	1.46E+03
F27	AVG	4.11E+03	3.55E+03	3.56E+03	3.58E+03	**3.48E+03**	3.54E+03	3.58E+03	4.38E+03
	STD	2.26E+02	7.66E+01	1.93E+02	8.07E+01	5.44E+01	**4.51E+01**	8.84E+01	5.21E+02
F28	AVG	**3.36E+03**	3.35E+03	3.38E+03	5.36E+03	3.35E+03	3.41E+03	4.21E+03	3.57E+03
	STD	3.98E+01	2.97E+01	4.30E+01	8.23E+02	1.83E+02	**2.68E+01**	3.17E+02	6.99E+01
F29	AVG	4.02E+03	4.27E+03	4.51E+03	3.91E+03	4.28E+03	**3.87E+03**	4.53E+03	5.98E+03
	STD	2.76E+02	3.35E+02	3.54E+02	2.33E+02	2.34E+02	**1.75E+02**	2.51E+02	5.34E+02
F30	AVG	**3.20E+03**	1.14E+06	9.90E+05	3.17E+07	3.18E+06	1.08E+06	1.08E+08	4.03E+07
	STD	**0.00E+00**	3.59E+05	4.69E+05	4.59E+07	2.39E+06	9.73E+04	4.12E+07	8.06E+06
W/T/L	AVG	**19/0/10**	0/0/29	2/0/27	3/0/26	3/0/26	2/0/27	0/0/29	0/0/29
	STD	**13/0/16**	0/0/29	1/0/28	1/0/28	1/0/28	**13/0/16**	0/0/0	0/0/0

configurations. A lower average Friedman ranking indicates better performance. In this regard, HHGATSD consistently ranks first (highlighted in blue) for both 30 and 50 dimensions, demonstrating its superior performance compared to all other algorithms in these tests. Holm's p-correction values help identify which algorithms exhibit statistically significant differences in their performance relative to others, providing a robust foundation for assessing the effectiveness of the algorithms in this study. Results in bold indicate p-values less than 0.05, denoting significant differences from HHGATSD.

TABLE 4. Average friedman's rankings and holm's *p* values (0.05) for the CEC2017 benchmark functions in 30 and 50 dimensions.

Algorithm	30D			50D		
	Ranking	No.	p_{Holm}	Ranking	No.	p_{Holm}
HHGATSD	2.2586	1		1.8793	1	
BSDE	3.0345	2	0.227769	3.5862	2	0.007967
DE	4.1552	3	**0.003195**	4.8966	6	**0.000003**
PSO	4.1897	4	**0.002683**	4.2069	3	**0.000296**
GA	4.4483	5	**0.000664**	4.3448	4	**0.000127**
JADE	5.3448	6	**0.000002**	4.6379	5	**0.000018**
GWO	5.7759	7	**0**	5.7241	7	**0**
HHO	6.7931	8	**0**	6.7241	8	**0**

Notably, the only algorithm with no significant difference in the results is the BSDE.

To graphically visualize the algorithm's performance, Figure 5 provides graphical representations of the convergence curves for all algorithms during a specific run of 50-dimensional instances. These curves, presented on a logarithmic scale for clarity, illustrate the convergence behavior of algorithms across different categories of benchmark functions.

It is worth noting that the convergence curves for HHGATSD (represented by a black square with a red dotted line), JADE (magenta triangle with a blue dotted line), BSDE (turquoise triangle with a turquoise dotted line), and GA (green star with a magenta dotted line) exhibit particularly favorable results. These algorithms demonstrate efficient convergence across various function categories, as evidenced by the minimized convergence curves.

In particular, HHGATSD consistently performs well when considering representative instances from different function categories such as unimodal (F1), multimodal (F4, F7, and F9), hybrid (F14 and F18), and composition (F28 and F30) functions.

B. EXPERIMENTS IN MULTILEVEL SEGMENTATION

This section presents the summary of the results of the experiments using the ten images from the BSDS500 and eight frames from the video traffic camera to address the multilevel thresholding segmentation.

The four criteria shown in the results in Table 5 are performance in terms of fitness and segmentation quality as measured by the PSNR, SSIM, and FSIM metrics for the BSDS500 images. These last three metrics provided a quantitative assessment of the performance of the algorithms in terms of segmentation accuracy and similarity to the original image.

The results presented in Table 5 are derived from non-parametric tests conducted on the AVG and STD values, extracted from 30 independent runs for each of the five thresholds applied to the images. The algorithms are ranked according to their average Friedman ranking, providing insights into their relative performance across each metric. Furthermore, Holm's p-correction values (denoted as *pHolm*) are included to determine statistical significance. These *pHolm* values represent adjusted p-values, accounting for multiple comparisons during the statistical analysis. Algorithms ranked first are highlighted in blue, while results with significant differences concerning HHGATSD are presented in bold.

Table 6 encompasses the computational results of the independently averaged over 30 independent runs for each of the five thresholds applied to the images, focusing on the ten images from the BSDS500 dataset and eight frames from the video traffic camera. In the BSDS500, JADE stands out for superior AVG and STD results, while the BSDE algorithm has the fastest computation times for the multilevel segmentation of the traffic video frames. Despite not taking first place, HHGATSD shows competitive performance with comparable average computation times, ranging from 0.56 to 0.95 seconds.

The proposed method, HH, is highly effective in thresholding and generates segmentations with high feature similarity to the original image, indicating remarkable accuracy. However, it only shows significant differences in fitness when compared with GA and GWO algorithms in terms of fitness and with DE, PSO, and GA algorithms in terms of FSIM. In PSNR, HHGATSD ranks third, with the JADE algorithm taking the first position. There was no significant difference between the algorithms for this criterion. Similarly, in SSIM, the HHGATSD ranks third, with the GWO algorithm taking the first position without any significant difference from the other algorithms, including our proposal. It is crucial to emphasize that, despite subtle variations, no statistically significant differences exist in computational times between the proposed HHGATSD and the compared algorithms, ensuring its viability for practical applications.

C. EXPERIMENTS WITH TRAFFIC VIDEO FRAMES FOR THE VEHICLE SEGMENTATION

In this section, we comprehensively analyze the HHGATSD algorithm's performance in vehicle segmentation on traffic video frames. Initially, we delve into hyperparameter combinations (*Iter* and α) for the multilevel segmentation of a subset of 61 images, aiming to identify the optimal hyperparameter values. Following this, we present visual representations of the significant steps of the methodology for segmenting eight frames. Subsequently, we evaluate and compare the performance of HHGATSD against other well-known segmentation techniques in the literature, considering all 612 images.

Table 7 focuses on analyzing combinations of hyperparameters (*Iter* and α) within the proposed HHGATSD algorithm. This study aims to experimentally identify and validate optimal hyperparameters for the algorithm's performance in multilevel segmentation of traffic video frames, serving as a preprocessing step for vehicle segmentation.

The experiments were conducted on 61 randomly selected images, representing 10% of the total dataset. The tests were done using 2, 4, 6, 8, and 10 thresholds to ensure a thorough evaluation of the algorithm's performance in vehicle segmentation for traffic video frames. The main objective of the research is to identify the most effective hyperparameter

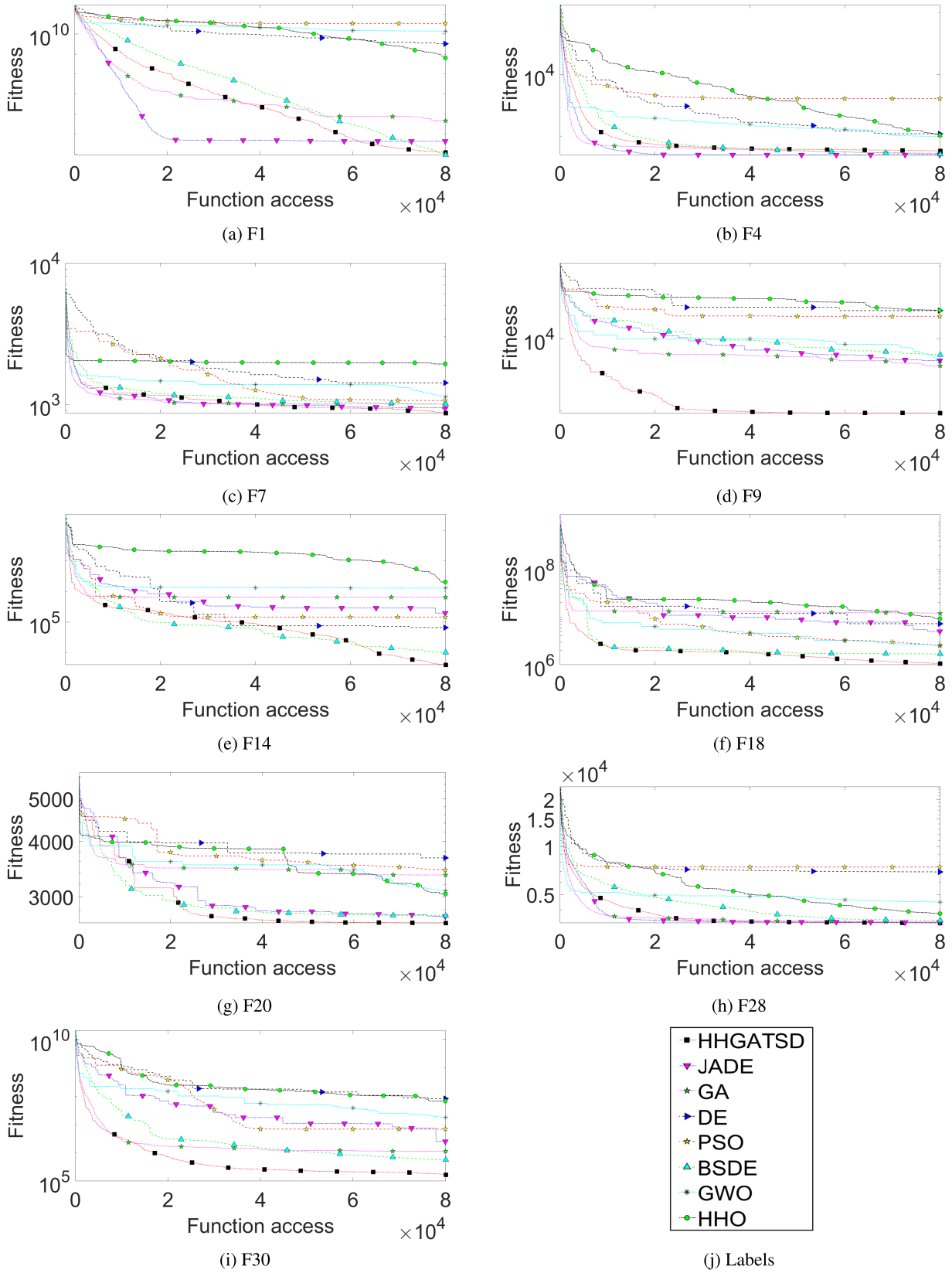


FIGURE 5. Convergence curves for all algorithms during a specific run of 50-dimensional instances. The curves employ a logarithmic scale for enhanced presentation quality.

TABLE 5. Average friedman's rankings and holm's *p* values (0.05) for the BSDS500.

Algorithm	Fitness			PSNR			SSIM			FSIM		
	Ranking	No.	p_{Holm}	Ranking	No.	p_{Holm}	Ranking	No.	p_{Holm}	Ranking	No.	p_{Holm}
HHGATSD	3.5667	**1**		4.35	3	0.654721	3.8833	3	0.314305	4.225	**1**	
BSDE	3.6833	2	0.794189	4.2833	2	0.765594	4.2667	4	0.062407	4.475	4	0.57615
DE	4.6667	5	0.013906	4.5167	5	0.412278	5.3333	7	**0.000022**	4.4333	3	0.641325
PSO	4.75	6	0.008145	4.8917	8	0.097233	5.0417	6	**0.000323**	4.7667	8	0.225817
GA	5.7583	7	**0.000001**	4.6333	6	0.279801	6.0333	8	**0**	4.6	7	0.401736
JADE	3.8167	4	0.57615	4.15	**1**		4.3833	5	0.033648	4.55	6	0.467396
GWO	5.9	8	**0**	4.75	7	0.179712	3.4333	**1**		4.5167	5	0.514281
HHO	3.8083	3	0.588933	4.425	4	0.538608	3.625	2	0.668229	4.4333	2	0.641325

TABLE 6. Average computation times (in seconds) for the BSDS500 and traffic video frames.

Dataset		BSDS500					Traffic video frames				
		2	4	6	8	10	2	4	6	8	10
HHGATSD	AVG	8.13E-01	8.73E-01	8.97E-01	9.25E-01	9.52E-01	5.74E-01	5.62E-01	6.03E-01	5.88E-01	6.01E-01
	STD	8.31E-02	7.63E-02	7.84E-02	7.93E-02	7.93E-02	8.95E-02	6.05E-02	8.58E-02	6.74E-02	6.06E-02
GA	AVG	3.12E-01	3.37E-01	3.64E-01	3.66E-01	3.64E-01	2.39E-01	2.33E-01	2.49E-01	**2.54E-01**	2.60E-01
	STD	2.63E-02	3.10E-02	3.04E-02	3.04E-02	2.74E-02	5.60E-02	2.89E-02	3.12E-02	2.91E-02	3.88E-02
PSO	AVG	3.73E-01	3.62E-01	3.78E-01	3.92E-01	4.14E-01	2.70E-01	2.81E-01	3.02E-01	3.03E-01	3.03E-01
	STD	3.71E-02	2.97E-02	3.24E-02	3.61E-02	3.53E-02	4.42E-02	3.20E-02	3.74E-02	4.60E-02	4.42E-02
DE	AVG	3.26E-01	3.41E-01	3.54E-01	3.71E-01	3.88E-01	2.43E-01	2.49E-01	2.58E-01	2.80E-01	2.84E-01
	STD	3.94E-02	3.06E-02	3.37E-02	3.25E-02	2.82E-02	4.47E-02	3.45E-02	3.09E-02	3.82E-02	5.02E-02
JADE	AVG	**2.07E-01**	**2.15E-01**	**2.35E-01**	**2.45E-01**	**2.48E-01**	3.01E-01	3.11E-01	3.25E-01	3.47E-01	3.41E-01
	STD	**2.11E-02**	**2.11E-02**	**2.12E-02**	**2.42E-02**	**2.38-02**	6.42E-02	3.61E-02	4.03E-02	4.82E-02	3.25E-02
BSDE	AVG	3.15E-01	3.31E-01	3.58E-01	3.84E-01	3.82E-01	**2.06E-01**	2.19E-01	**2.31E-01**	**2.40E-01**	**2.55E-01**
	STD	4.17E-02	3.29E-02	3.50E-02	4.15E-02	3.49E-02	**3.46E-02**	3.21E-02	3.16E-02	2.95E-02	3.49E-02
GWO	AVG	2.78E-01	2.92E-01	3.39E-01	3.73E-01	3.84E-01	2.08E-01	**2.14E-01**	2.44E-01	2.68E-01	2.74E-01
	STD	3.55E-02	2.99E-02	3.29E-02	3.51E-02	4.06E-02	4.19E-02	**2.73E-02**	3.30E-02	3.93E-02	**2.75E-02**
HHO	AVG	6.86E-01	7.32E-01	7.96E-01	8.31E-01	8.59E-01	2.56E-01	2.63E-01	2.78E-01	3.14E-01	3.06E-01
	STD	9.54E-02	8.89E-02	9.18E-02	7.67E-02	1.01E-01	3.79E-02	3.23E-02	**2.86E-02**	4.90E-02	3.32E-02

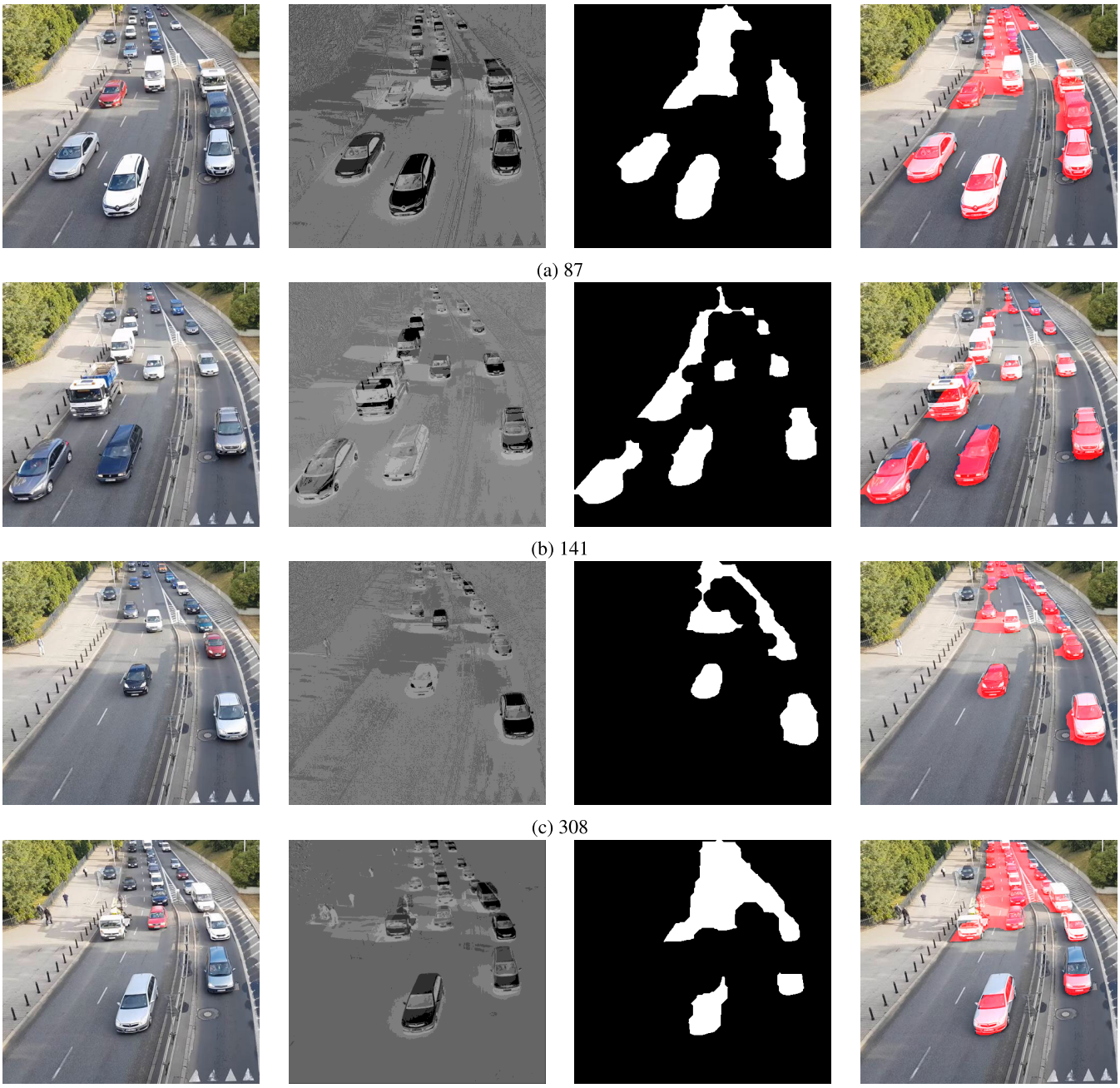
TABLE 7. Stadistical results and average friedman's rankings and holm's *p* values (0.05) for traffic video frames.

Iter	α	Ranking	No.	p_{Holm}	Fitness		PSNR		SSIM		FSIM	
					AVG	STD	AVG	STD	AVG	STD	AVG	STD
50	1	8.375	2	0.5524	2.17E-01	6.10E-04	2.49E+01	5.58E-02	8.67E-01	1.37E-03	9.48E-01	5.82E-04
	3	10.375	3	0.2347	2.18E-01	8.03E-04	2.49E+01	7.24E-02	8.67E-01	1.63E-03	9.48E-01	7.75E-04
	5	10.375	3	0.2347	2.18E-01	8.78E-04	2.49E+01	7.76E-02	8.67E-01	1.73E-03	9.48E-01	7.91E-04
	10	12.375	4	0.0747	2.18E-01	1.01E-03	2.49E+01	7.71E-02	8.67E-01	1.73E-03	9.47E-01	7.21E-04
100	1	6.375	**1**		2.17E-01	5.61E-04	2.49E+01	4.72E-02	8.68E-01	1.20E-03	9.48E-01	5.12E-04
	3	8.375	2	0.5524	2.17E-01	7.49E-04	2.49E+01	6.60E-02	8.67E-01	1.54E-03	9.48E-01	6.69E-04
	5	8.375	2	0.5524	2.17E-01	8.27E-04	2.49E+01	6.99E-02	8.67E-01	1.57E-03	9.48E-01	6.97E-04
	10	10.375	3	0.2347	2.18E-01	9.22E-04	2.49E+01	6.81E-02	8.67E-01	1.59E-03	9.48E-01	6.64E-04
200	1	6.375	**1**	1	2.17E-01	**5.12E-04**	2.49E+01	4.21E-02	8.68E-01	1.09E-03	9.48E-01	4.23E-04
	3	8.375	2	0.5524	2.17E-01	6.31E-04	2.49E+01	5.09E-02	8.67E-01	1.28E-03	9.48E-01	5.32E-04
	5	8.375	2	0.5524	2.17E-01	7.71E-04	2.49E+01	5.69E-02	8.67E-01	1.38E-03	9.48E-01	5.43E-04
	10	8.375	2	0.5524	2.17E-01	8.69E-04	2.49E+01	6.05E-02	8.67E-01	1.46E-03	9.48E-01	5.78E-04
300	1	6.375	**1**	1	**2.17E-01**	5.16E-04	2.49E+01	**4.08E-02**	**8.68E-01**	**1.04E-03**	**9.48E-01**	**3.87E-04**
	3	6.375	**1**	1	2.17E-01	6.60E-04	2.49E+01	5.12E-02	8.68E-01	1.24E-03	9.48E-01	4.93E-04
	5	8.375	2	0.5524	2.17E-01	7.80E-04	**2.49E+01**	5.33E-02	8.67E-01	1.37E-03	9.48E-01	5.29E-04
	10	8.375	2	0.5524	2.17E-01	7.66E-04	2.49E+01	5.26E-02	8.67E-01	1.37E-03	9.48E-01	4.98E-04

values for vehicle segmentation. The table provides detailed rankings and Holm's p-values obtained from non-parametric Friedman tests. Furthermore, statistical results for AVG and STD in fitness, PSNR, SSIM, and FSIM are presented, with the best AVG highlighted in blue and the highest STD in bold.

Upon analyzing the data, it becomes evident that there are no significant differences among various hyperparameter configurations, indicated by the lack of bold Holm's p-values (all above 0.05). This observation underscores the consistency of the algorithm's performance across different hyperparameter combinations. Notably, the algorithm maintains stability and reliability in its performance, even with variations in internal parameters like *Iter* and α. The average values across all combinations exhibit no noteworthy distinctions, underscoring the algorithm's versatility and effectiveness in tackling the vehicle segmentation challenge.

The conducted experiments confirm that the chosen hyperparameter values (*Iter* $=$ 200 and α $=$ 5) remain stable and effective for the vehicle segmentation task in traffic video frames. These values consistently yield satisfactory



(a) 87

(b) 141

(c) 308

(d) 389

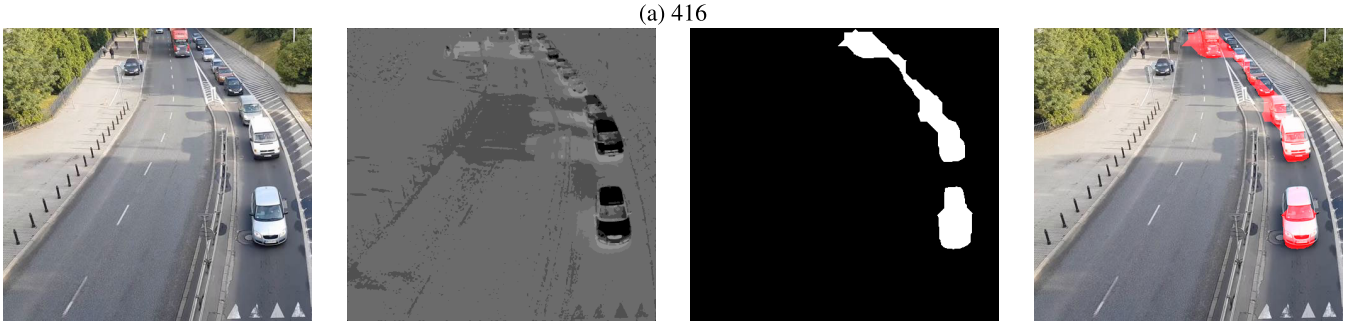
FIGURE 6. *Visual Examples of the Proposed Methodology.*

results, as demonstrated by both statistical analyses and the absence of significant performance differences across various hyperparameter configurations.

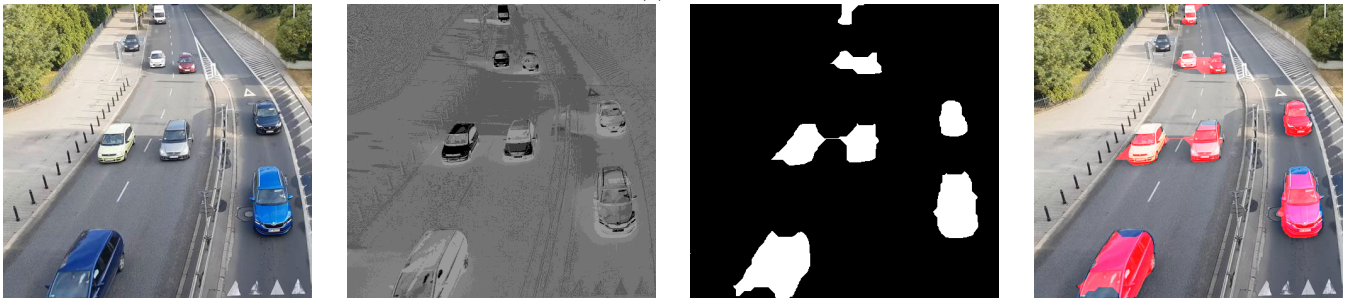
Table 8 presents a comprehensive overview of the AVG performance metrics alongside their respective STD values for the methodology for vehicle segmentation. The values highlighted in blue indicate the algorithms with the best AVG performance for the evaluated metrics, while bold values signify the results with the minimum STD. Notably, for DSC and IoU metrics, the proposed HHGATSD emerges as the leader, achieving AVG scores of 0.82 and 0.70, respectively. It also demonstrates the lowest STD values for these metrics.

Concerning the P metric, K-means emerges as the top-performing algorithm with an AVG value of 0.89 and a STD of 0.12. However, it is worth noting that HHGATSD closely follows with an AVG of 0.87 and a STD of 0.13, demonstrating only a slight variation between the two. For the R and ACC metrics, HHGATSD once again showcases the highest AVG values and the smallest STD values among all algorithms tested.

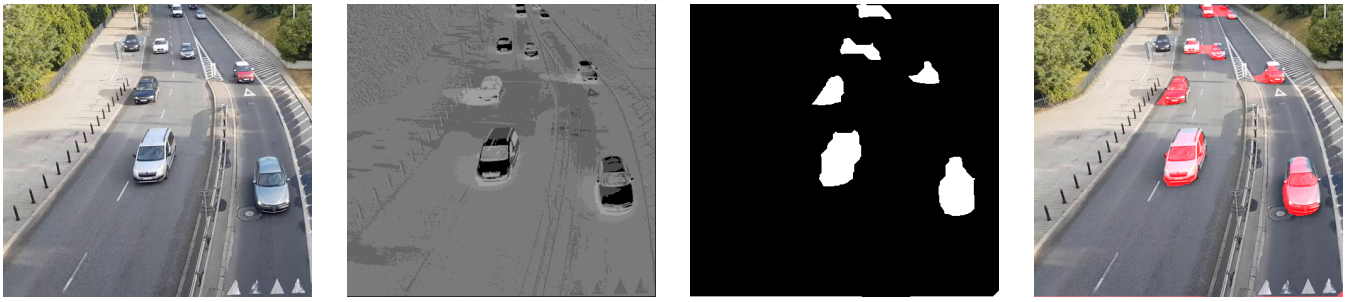
(a) 416



(b) 441



(c) 507



(d) 534

FIGURE 7. Visual Examples of the Proposed Methodology.

Regarding computational time, it is essential to highlight that Otsu achieved the fastest segmentation time, with an average of 0.46 seconds per image. HHGATSD demonstrated notable efficiency in close succession with an average of 0.49 seconds. Although K-means exhibited a slightly higher average of 0.50 seconds, Fuzzy IterAg presented significantly higher segmentation time, with an average of 5.58 seconds per image. Importantly, the observed differences in computational time between the proposed HHGATSD approach and alternative algorithms, such as Otsu and K-means, do not indicate any statistically significant differences. This underscores the efficiency of our methodology for vehicle segmentation.

Figures 6 and 7 offer an in-depth illustration of the effectiveness of our proposed methodology, utilizing the HHGATSD algorithm for multilevel segmentation to segment vehicles in the traffic video sequence. Each of the eight frames represents a different scenario, with each row representing the process of obtaining the mask for each one. Meanwhile, the columns present a series of key images, showcasing the most critical stages of our proposed methodology. This visual representation illustrates the

TABLE 8. Statistical results of the performance metrics in vehicle segmentation.

Metric		HHGATSD	Otsu	K-means	Fuzzy IterAg
DSC	AVG	**0.82**	0.74	0.76	0.73
	STD	**0.08**	0.13	0.10	0.15
IoU	AVG	**0.70**	0.60	0.62	0.60
	STD	**0.12**	0.15	**0.12**	0.16
P	AVG	0.87	0.88	**0.89**	0.83
	STD	0.13	**0.12**	**0.12**	0.22
R	AVG	**0.80**	0.66	0.68	0.72
	STD	**0.11**	0.16	0.13	0.13
ACC	AVG	**0.96**	0.95	0.95	0.95
	STD	**0.02**	0.03	0.03	0.03
Time	AVG	0.49	**0.46**	0.50	5.58
	STD	0.12	**0.03**	0.09	1.45

gradual transformation of the original image into a mask that segments the vehicles.

The first column displays the original $\mathbf{I_{th}}$ image of the traffic scene captured by the surveillance camera. The second column reveals the component image L, acquired through the RobustPCA process, which captures the image's underlying structural information. This component undergoes multilevel segmentation with five thresholds ($\mathbf{I_{th}}$).

In the third column, we present the mask obtained by applying binarization and morphological operations ($\mathbf{I_{seg}}$). This mask accentuates the regions corresponding to the vehicles within the frames. Finally, the fourth column displays the original image once more, overlaying the final segmentation mask ($\mathbf{I_{seg}}$) in red. This red overlay facilitates the identification of segmented areas.

It is important to mention that as vehicles draw closer to the camera, their spatial distance increases, leading to a proportional increase in the number of pixels they occupy. Consequently, in such scenarios, the segmentation becomes more refined and accurate, capturing more delimited details with greater precision. On the other hand, it is worth emphasizing that due to the method's capability to perform segmentation without prior background or vehicle-labeled information, automobiles positioned far from the camera are not delineated as separate entities. Still, they contribute significantly to understanding traffic dynamics, especially vehicular density. Thus, despite not being isolated, their presence is vital for comprehensive traffic analysis.

VII. CONCLUSION

This work presents a novel HH approach, including a method for segmenting vehicles in video frames captured by traffic cameras, which were rigorously tested through three comprehensive sets of experiments.

The study commences by subjecting HHGATSD to a rigorous evaluation, testing it on 29 functions from the IEEE CEC2017 benchmark for optimization problems. The results consistently showcase its robustness and competitive performance in 30 and 50 dimensions, surpassing seven state-of-the-art optimization algorithms. Results validate the proposal's performance through convergence graph analysis and Friedman's non-parametric statistical test, establishing it as the highest-ranked algorithm for both dimensions.

Next, an experiment focused on image segmentation with implications for object recognition and scene analysis was applied. Multilevel segmentation was applied to ten random images selected from the BSDS500. In these experiments, 2, 4, 6, 6, 8, and 10 thresholds were used, in which the proposed approach consistently outperformed the other optimization algorithms in terms of fitness, PSNR, SSIM, and FSIM. Notably, it secures the top-rank in Friedman's non-parametric test for objective function minimization and the FSIM metric, attesting to its efficacy in accurately delineating object boundaries in various imaging contexts. However, due to the low dimensionality of the problems used, it did not exhibit significant differences in most cases during the Holm test.

The culmination of our research trials involves the application of our methodology to vehicle segmentation in traffic video sequences, yielding promising results. HHGATSD plays a pivotal role in multilevel image segmentation, outperforming three commonly used techniques in overlap metrics by comparing the obtained mask against ground truth, particularly in DSC, IoU, R, and ACC. These results underscore its suitability for moving object segmentation in complex traffic environments, and its ability to maintain low STD values in these metrics highlights its consistency, a critical aspect in object tracking applications.

Experiments show that HHGATSD is a promising HH for complex problems such as multilevel image segmentation. Furthermore, the methodology produced accurate vehicle segmentation in traffic camera video. Additionally, the results of this study have opened up various directions for future research, which we believe are critical for furthering the use and refining of this methodology. As further work, the study is intended to continue by strategically deploying an additional camera to measure road occupancy percentages and obtain a numerical car count. With this information, we would be able to identify areas of high traffic density and make data-driven decisions to reduce congestion and optimize traffic flow, ultimately improving the overall efficiency of the transportation system.

REFERENCES

[1] N. Boysen, S. Fedtke, and S. Schwerdfeger, "Last-mile delivery concepts: A survey from an operational research perspective," *Or Spectr.*, vol. 43, pp. 1–58, Mar. 2021.

[2] T. Bosona, "Urban freight last mile logistics—Challenges and opportunities to improve sustainability: A literature review," *Sustainability*, vol. 12, no. 21, p. 8769, 2020.

[3] S. M. Patella, G. Grazieschi, V. Gatta, E. Marcucci, and S. Carrese, "The adoption of green vehicles in last mile logistics: A systematic review," *Sustainability*, vol. 13, no. 1, p. 6, 2020.

[4] Á. Halldórsson and J. Wehner, "Last-mile logistics fulfilment: A framework for energy efficiency," *Res. Transp. Bus. Manage.*, vol. 37, p. 100481, Dec. 2020.

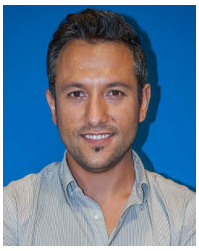
[5] S. Sorooshian, S. Khademi Sharifabad, M. Parsaee, and A. R. Afshari, "Toward a modern last-mile delivery: Consequences and obstacles of intelligent technology," *Appl. Syst. Innov.*, vol. 5, no. 4, p. 82, 2022.

[6] E. Demir, A. Syntetos, and T. V. Woensel, "Last mile logistics: Research trends and needs," *IMA J. Manage. Math.*, vol. 33, no. 4, pp. 549–561, 2022.
[7] N. Al Suwaidi, "Integrating all mobility systems to reduce traffic congestion," 2020.
[8] E. Namazi, J. Li, and C. Lu, "Intelligent intersection management systems considering autonomous vehicles: A systematic literature review," *IEEE Access*, vol. 7, pp. 91946–91965, 2019.
[9] V. C. SS and A. HS, "Nature inspired meta heuristic algorithms for optimization problems," *Computing*, vol. 104, no. 2, pp. 251–269, 2022.
[10] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. MHS 6th Int. Symp. Micro Mach. Human Sci.*, Oct. 1995, pp. 39–43, doi: 10.1109/MHS.1995.494215.
[11] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.
[12] O. Maciel, E. Cuevas, M. A. Navarro, D. Zaldívar, and S. Hinojosa, "Side-blotched lizard algorithm: A polymorphic population approach," *Appl. Soft Comput.*, vol. 88, p. 106039, Mar. 2020.
[13] J. H. Holland, "Genetic algorithms," *Sci. Amer.*, vol. 267, no. 1, pp. 66–73, 1992.
[14] J. Zhang and A. C. Sanderson, "JADE: Self-adaptive differential evolution with fast and reliable convergence performance," in *Proc. IEEE Congr. Evol. Comput.*, Sep. 2007, pp. 2251–2258.
[15] P. Civicioglu and E. Besdok, "Bernstain-search differential evolution algorithm for numerical function optimization," *Expert Syst. Appl.*, vol. 138, p. 112831, Dec. 2019.
[16] X. Chen, H. Tianfield, C. Mei, W. Du, and G. Liu, "Biogeography-based learning particle swarm optimization," *Soft Comput.*, vol. 21, pp. 7519–7541, Dec. 2017.
[17] A. A. Heidari, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019, doi: 10.1016/j.future.2019.02.028. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X18313530
[18] H. Jia, X. Peng, and C. Lang, "Remora optimization algorithm," *Expert Syst. Appl.*, vol. 185, p. 115665, Dec. 2021.
[19] K. Avila, E. Cuevas, M. Perez, and R. Sarkar, "A new metaphor-free metaheuristic approach based on complex networks and Bezier curves," *IEEE Access*, vol. 11, pp. 1–32, 2023.
[20] M. A. Navarro, G. R. Hernández, D. Zaldívar, N. Ortega-Sanchez, and G. Pajares, "Segmentation of thermal images using metaheuristic algorithms for failure detection on electronic systems," *Appl. Hybrid metaheuristic Algorithms Image Process.*, pp. 3–26, 2020.
[21] E. Cuevas, A. Rodríguez, A. Alejo-Reyes, C. Del-Valle-Soto, E. Cuevas, A. Rodríguez, A. Alejo-Reyes, and C. Del-Valle-Soto, "Blood vessel segmentation using differential evolution algorithm," *Recent metaheuristic Comput. Schemes Eng.*, pp. 151–167, 2021.
[22] O. Ramos-Soto, E. Rodríguez-Esparza, S. E. Balderas-Mata, D. Oliva, A. E. Hassanien, R. K. Meleppat, and R. J. Zawadzki, "An efficient retinal blood vessel segmentation in eye fundus images by using optimized top-hat and homomorphic filtering," *Comput. Methods Programs Biomed.*, vol. 201, Apr. 2021, Art. no. 105949.
[23] P. J. Herrera, F. Montes, G. Pajares, and C. Cerrada, "An automatic segmentation method based on geometrical features in hemispherical images for forest management," in *Proc. 12th Int. Conf. Digit. Image Process. (ICDIP)*, Jun. 2020, pp. 146–154.
[24] H. Jia, C. Lang, D. Oliva, W. Song, and X. Peng, "Dynamic Harris hawks optimization with mutation mechanism for satellite image segmentation," *Remote Sens.*, vol. 11, no. 12, p. 1421, Jun. 2019.
[25] T. Dutta, S. Bhattacharyya, B. K. Panigrahi, and A. E. Hassanien, "Automatic clustering of hyperspectral images using quantum reptile search algorithm," in *Proc. Int. Conf. Variability Sun-Like Stars, Astero-seismology Space Weather*. Springer, 2022, pp. 653–664.
[26] A. Swiercz, "Hyper-heuristics and metaheuristic s for selected bio-inspired combinatorial optimization problems," in *Heuristics Hyper-Heuristics—Princ. Appl.*, 2017.
[27] E. K. Burke, M. Gendreau, M. Hyde, G. Kendall, G. Ochoa, E. Özcan, and R. Qu, "Hyper-heuristics: A survey of the state of the art," *J. Oper. Res. Soc.*, vol. 64, no. 12, pp. 1695–1724, Dec. 2013.
[28] J. C. Gomez and H. Terashima-Marín, "Evolutionary hyper-heuristics for tackling bi-objective 2D bin packing problems," *Genetic Program. Evolvable Mach.*, vol. 19, nos. 1–2, pp. 151–181, Jun. 2018.
[29] S. van der Stockt and A. P. Engelbrecht, "Analysis of hyper-heuristic performance in different dynamic environments," in *Proc. IEEE Symp. Comput. Intell. Dyn. Uncertain Environments (CIDUE)*, Dec. 2014, pp. 1–8.
[30] M. Lassouaoui and D. Boughaci, "A choice function hyper-heuristic for the winner determination problem," *Nature Inspired Cooperat. Strategies Optim., Learn., Optim. Interdiscipl. Appl.*, pp. 303–314, 2013.
[31] E. S. Sin and N. S. M. Kham, "Hyper heuristic based on great deluge and its variants for exam timetabling problem," 2012, *arXiv:1202.1891*.
[32] A. S. Ferreira, R. A. Goncalves, and A. Trinidad Ramirez Pozo, "A multi-armed bandit hyper-heuristic," in *Proc. Brazilian Conf. Intell. Syst. (BRACIS)*, Nov. 2015, pp. 13–18.
[33] E. K. Burke, T. Curtois, M. Hyde, G. Kendall, G. Ochoa, S. Petrovic, and J. A. Vázquez-Rodríguez, "Hyflex: A flexible framework for the design and analysis of hyper-heuristics," in *Proc. Multidisciplinary Int. Scheduling Conf.*, 2009, pp. 790–797.
[34] F. Zhao, S. Di, J. Cao, and J. Tang, "A novel cooperative multi-stage hyper-heuristic for combination optimization problems," *Complex Syst. Model. Simul.*, vol. 1, no. 2, pp. 91–108, Jun. 2021.
[35] Y. Sun, L. Jia, and Z. Zeng, "Hyper-heuristic capacitance array method for multi-metal wear debris detection," *Sensors*, vol. 19, no. 3, p. 515, Jan. 2019.
[36] S. Mahmud, A. Abbasi, R. K. Chakrabortty, and M. J. Ryan, "A self-adaptive hyper-heuristic based multi-objective optimisation approach for integrated supply chain scheduling problems," *Knowl.-Based Syst.*, vol. 251, Sep. 2022, Art. no. 109190.
[37] W. Qin, Z. Zhuang, Z. Huang, and H. Huang, "A novel reinforcement learning-based hyper-heuristic for heterogeneous vehicle routing problem," *Comput. Ind. Eng.*, vol. 156, Jun. 2021, Art. no. 107252.
[38] M. A. Elaziz and S. Mirjalili, "A hyper-heuristic for improving the initial population of whale optimization algorithm," *Knowl.-Based Syst.*, vol. 172, pp. 42–63, May 2019.
[39] S. El-Khatib, Y. Skobtsov, S. Rodzin, and V. Zelentsov, "Hyper-heuristical particle swarm method for mr images segmentation," in *Proc. Artif. Intell. Algorithms Intell. Syst. Proc. 7th Comput. Sci. Line Conf.*, vol. 2. Springer, 2018, pp. 256–264.
[40] M. A. Elaziz, A. A. Ewees, and D. Oliva, "Hyper-heuristic method for multilevel thresholding image segmentation," *Expert Syst. Appl.*, vol. 146, May 2020, Art. no. 113201.
[41] F. Jafari Nozar and J. Behnamian, "Hyper-heuristic for integrated due-window scheduling and vehicle routing problem for perishable products considering production quality," *Eng. Optim.*, vol. 53, no. 11, pp. 1902–1921, Nov. 2021.
[42] C. Zhang, N. R. Sabar, E. Chung, A. Bhaskar, and X. Guo, "Optimisation of lane-changing advisory at the motorway lane drop bottleneck," *Transp. Res. Part C, Emerg. Technol.*, vol. 106, pp. 303–316, Sep. 2019.
[43] P. Heyken Soares, L. Ahmed, Y. Mao, and C. L. Mumford, "Public transport network optimisation in PTV visum using selection hyper-heuristics," *Public Transp.*, vol. 13, no. 1, pp. 163–196, Mar. 2021.
[44] A. Tarhini, K. Danach, and A. Harfouche, "Swarm intelligence-based hyper-heuristic for the vehicle routing problem with prioritized customers," *Ann. Operations Res.*, vol. 308, nos. 1–2, pp. 549–570, Jan. 2022.
[45] E. Rodríguez-Esparza, A. D. Masegosa, D. Oliva, and E. Onieva, "A new hyper-heuristic based on adaptive simulated annealing and reinforcement learning for the capacitated electric vehicle routing problem," 2022, *arXiv:2206.03185*.
[46] P. B. Prakoso and Y. Sari, "Vehicle detection using background subtraction and clustering algorithms," *TELKOMNIKA (Telecommun. Comput. Electron. Control)*, vol. 17, no. 3, p. 1393, Jun. 2019.
[47] C. Premachandra, S. Ueda, and Y. Suzuki, "Detection and tracking of moving objects at road intersections using a 360-degree camera for driver assistance and automated driving," *IEEE Access*, vol. 8, pp. 135652–135660, 2020.
[48] C. Premachandra, D. Ueda, and K. Kato, "Speed-up automatic quadcopter position detection by sensing propeller rotation," *IEEE Sensors J.*, vol. 19, no. 7, pp. 2758–2766, Apr. 2019.
[49] X. Hu, X. Xu, Y. Xiao, H. Chen, S. He, J. Qin, and P.-A. Heng, "SINet: A scale-insensitive convolutional neural network for fast vehicle detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 1010–1019, Mar. 2019.
[50] V. S. Sindhu, "Vehicle identification from traffic video surveillance using YOLOV4," in *Proc. 5th Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, May 2021, pp. 1768–1775.

[51] A. Kashevnik and A. Ali, "3D vehicle detection and segmentation based on EfficientNetB3 and CenterNet residual blocks," *Sensors*, vol. 22, no. 20, p. 7990, Oct. 2022.
[52] G. Cheng, C. Lang, and J. Han, "Holistic prototype activation for few-shot segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 4650–4666, Apr. 2023.
[53] C. Lang, G. Cheng, B. Tu, C. Li, and J. Han, "Base and meta: A new perspective on few-shot segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–18, Apr. 2023.
[54] A. Khassiba and D. Delahaye, "Simulated-annealing hyper-heuristic for demand-capacity balancing in air traffic flow management," *SESAR Innov. Days*, 2022.
[55] X.-C. Liao, Y.-H. Jia, X.-M. Hu, and W.-N. Chen, "Uncertain commuters assignment through genetic programming hyper-heuristic," *IEEE Trans. Computat. Social Syst.*, pp. 1–14, Apr. 2023.
[56] Z. Wang, J. Liu, and J. Zhang, "Hyper-heuristic algorithm for traffic flow-based vehicle routing problem with simultaneous delivery and pickup," *J. Comput. Design Eng.*, vol. 10, no. 6, pp. 2271–2287, Nov. 2023.
[57] X. Huang, D. Mu, and Z. Li, "Intelligent traffic analysis: A heuristic high-dimensional image search algorithm based on spatiotemporal probability for constrained environments," *Alexandria Eng. J.*, vol. 59, no. 3, pp. 1413–1423, Jun. 2020.
[58] C. S. Priya and F. S. Francis, "A novel LoRaWAn-based real-time traffic analysis approach for vehicle congestion estimation," in *Proc. Winter Summit Smart Comput. Netw. (WiSSCoN)*, Mar. 2023, pp. 1–6.
[59] E. Rodríguez-Esparza, L. A. Zanella-Calzada, D. Oliva, A. A. Heidari, D. Zaldivar, M. Pérez-Cisneros, and L. K. Foong, "An efficient Harris hawks-inspired image segmentation method," *Expert Syst. Appl.*, vol. 155, Oct. 2020, Art. no. 113428.
[60] P.-Y. Yin, "Multilevel minimum cross entropy threshold selection based on particle swarm optimization," *Appl. Math. Comput.*, vol. 184, no. 2, pp. 503–513, Jan. 2007.
[61] M. Scoczynski, D. Oliva, E. Rodríguez-Esparza, M. Delgado, R. Lüders, M. E. Yafrani, L. Ledo, M. A. Elaziz, and M. Peréz-Cisnero, "A selection hyperheuristic guided by Thompson sampling for numerical optimization," in *Proc. Genetic Evol. Comput. Conf. Companion*, Jul. 2021, pp. 1394–1402.
[62] G. Corriveau, R. Guilbault, A. Tahan, and R. Sabourin, "Review and study of genotypic diversity measures for real-coded representations," *IEEE Trans. Evol. Comput.*, vol. 16, no. 5, pp. 695–710, Oct. 2012.
[63] A. Shukla, H. M. Pandey, and D. Mehrotra, "Comparative review of selection techniques in genetic algorithm," in *Proc. Int. Conf. Futuristic Trends Comput. Anal. Knowl. Manage. (ABLAZE)*, Feb. 2015, pp. 515–519.
[64] A. Slivkins, "Introduction to multi-armed bandits," *Found. Trends Mach. Learn.*, vol. 12, nos. 1–2, pp. 1–286, 2019.
[65] S. De, A. Ghosh, and S. K. Pal, "Fitness evaluation in genetic algorithms with Ancestors' influence," in *Genetic Algorithms Pattern Recognition*. Boca Raton, FL, USA: CRC Press, 2017, pp. 1–24.
[66] D. Oliva, E. Rodriguez-Esparza, M. S. R. Martins, M. A. Elaziz, S. Hinojosa, A. A. Ewees, and S. Lu, "Balancing the influence of evolutionary operators for global optimization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2020, pp. 1–8.
[67] I. Ono and S. Kobayashi, "A real-coded genetic algorithm for function optimization using unimodal normal distribution," in *Proc. Int. Conf. Genetic Algorithms*, 1999, pp. 246–253.
[68] K. Deb, D. Joshi, and A. Anand, "Real-coded evolutionary algorithms with parent-centric recombination," in *Proc. Congr. Evol. Comput.*, 2002, pp. 61–66.
[69] L. J. Eshelman and J. D. Schaffer, "Real-coded genetic algorithms and interval-schemata," in *Foundations of Genetic Algorithms*. Amsterdam, The Netherlands: Elsevier, 1993, pp. 187–202.
[70] R. Poli and W. B. Langdon, "Schema theory for genetic programming with one-point crossover and point mutation," *Evol. Comput.*, vol. 6, no. 3, pp. 231–252, Sep. 1998.
[71] S. C. Esquivel and C. A. Coello Coello, "On the use of particle swarm optimization with multimodal functions," in *Proc. Congr. Evol. Comput.*, 2003, pp. 1130–1136.
[72] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.
[73] I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Inf. Sci.*, vol. 237, pp. 82–117, Jul. 2013.
[74] H. R. Tizhoosh, "Opposition-based learning: A new scheme for machine intelligence," in *Proc. Int. Conf. Comput. Intell. Model., Control Autom. Int. Conf. Intell. Agents, Web Technol. Internet Commerce*, 2005, pp. 695–701.
[75] G. Wu, R. Mallipeddi, and P. N. Suganthan, "Problem definitions and evaluation criteria for the cec 2017 competition on constrained real-parameter optimization," *Nat. Univ. Defense Technol., Changsha, Hunan, PR China Kyungpook Nat. Univ., Daegu, South Korea Nanyang Technological Univ., Singap.*, Tech. Rep., 2017.
[76] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011, doi: 10.1109/TPAMI.2010.161. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2010.161
[77] F. A. Hashim, E. H. Houssein, K. Hussain, M. S. Mabrouk, and W. Al-Atabany, "Honey badger algorithm: New metaheuristic algorithm for solving optimization problems," *Math. Comput. Simul.*, vol. 192, pp. 84–110, Feb. 2022.
[78] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.
[79] C. Saha and Md. F. Hossain, "MRI brain tumor images classification using K-means clustering, NSCT and SVM," in *Proc. 4th IEEE Uttar Pradesh Sect. Int. Conf. Electr., Comput. Electron. (UPCON)*, Oct. 2017, pp. 329–333.
[80] S. Aja-Fernández, A. H. Curiale, and G. Vegas-Sánchez-Ferrero, "A local fuzzy thresholding methodology for multiregion image segmentation," *Knowl.-Based Syst.*, vol. 83, pp. 1–12, Jul. 2015.
[81] P. Ghamisi, M. S. Couceiro, J. A. Benediktsson, and N. M. F. Ferreira, "An efficient method for segmentation of images based on fractional calculus and natural selection," *Expert Syst. Appl.*, vol. 39, no. 16, pp. 12407–12417, Nov. 2012.
[82] A. Horé and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 2366–2369.
[83] D. R. I. M. Setiadi, "PSNR vs SSIM: Imperceptibility quality assessment for image steganography," *Multimedia Tools Appl.*, vol. 80, no. 6, pp. 8423–8444, Mar. 2021.
[84] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.
[85] E. R. Arce-Santana, A. R. Mejia-Rodriguez, E. Martinez-Peña, A. Alba, M. Mendez, E. Scalco, A. Mastropietro, and G. Rizzo, "A new probabilistic active contour region-based method for multiclass medical image segmentation," *Med. Biol. Eng. Comput.*, vol. 57, no. 3, pp. 565–576, Mar. 2019.
[86] J. Zhang, J. Du, H. Liu, X. Hou, Y. Zhao, and M. Ding, "LU-NET: An improved U-Net for ventricular segmentation," *IEEE Access*, vol. 7, pp. 92539–92546, 2019.
[87] O. J. Bastidas, B. Garcia-Zapirain, A. L. Totoricagüena, S. Zahia, and J. U. Carpio, "Feature analysis and prediction of complications in ostomy patients based on laboratory analytical data using a machine learning approach," in *Proc. Int. Conf. BIOMDLORE*, Oct. 2021, pp. 1–8.
[88] O. Jossa-Bastidas, S. Zahia, A. Fuente-Vidal, N. Sánchez Férez, O. Roda Noguera, J. Montane, and B. Garcia-Zapirain, "Predicting physical exercise adherence in fitness apps using a deep learning approach," *Int. J. Environ. Res. Public Health*, vol. 18, no. 20, p. 10769, Oct. 2021.
[89] M. Scoczynski, M. Delgado, R. Lüders, D. Oliva, M. Wagner, I. Sung, and M. El Yafrani, "Saving computational budget in Bayesian network-based evolutionary algorithms," *Natural Comput.*, vol. 20, no. 4, pp. 775–790, Dec. 2021.
[90] N. A. Aziz, Z. Ibrahim, S. Razali, and N. A. A. Aziz, "Estimation-based metaheuristic s: A new branch of computational intelligence," in *Proc. Nat. Conf. Postgraduate Res.*, 2016, pp. 469–476.

ERICK RODRÍGUEZ-ESPARZA received the B.S. degree in biomedical engineering from the University of San Luis Potosi, Mexico, in 2017, and the M.Sc. degree in electronic engineering and computer sciences from the University of Guadalajara, Mexico, in 2020. He is currently pursuing the Ph.D. degree in engineering for the information society and sustainable development with the University of Deusto, Spain. His research interests include evolutionary and swarm algorithms, digital image processing, machine and deep learning, biomedical engineering, and intelligent transportation systems.

OSCAR RAMOS-SOTO received the B.S. degree in communications and electronics engineering from the National Polytechnic Institute, Mexico, in 2019, and the M.Sc. degree in electronic engineering and computer sciences from the University of Guadalajara, Mexico, in 2021, where he is currently pursuing the Ph.D. degree in electronics and computer sciences. His research interests include computer vision, digital image processing, machine and deep learning, biomedical engineering, and vision science.



ANTONIO D. MASEGOSA received the degree in computer engineering and the Ph.D. degree in computer sciences from the University of Granada, Spain, in 2005 and 2010, respectively. From June 2010 to November 2014, he was a Postdoctoral Researcher with the Research Center for ICT, University of Granada. In 2014, he received an Ikerbasque Research Fellowship to work with the Mobility Unit, DeustoTech, Bilbao, Spain. He is a Principal Investigator of the Deusto Smart Mobility Group. He has led several research projects at regional, national, and European levels. He has published four books, 25 JCR papers, and more than 30 papers at both international and national conferences. His main research interests include artificial intelligence, intelligent systems, soft computing, hybrid metaheuristics, machine learning, deep learning, intelligent transportation systems, logistic networks, travel behavior analysis, traffic forecasting, and traffic accident prediction. In 2019, he was awarded the Ikerbasque Research Associate.

ENRIQUE ONIEVA received the B.E. degree in computer science engineering, the M.E. degree in soft computing and intelligent systems, and the Ph.D. degree in computer science from the University of Granada, Spain, in 2006, 2008, and 2011, respectively.

From 2007 to 2012, he was with the AUTOPIA Program, Centre of Automation and Robotics, Consejo Superior de Investigaciones Científicas, Madrid, Spain. In 2012, he joined the Models of Decision and Optimization Group, University of Granada. Since 2013, he has been a Professor of artificial intelligence with the University of Deusto and a Researcher of intelligent transportation systems applications with the Deusto Smart Mobility Research Unit. He has participated in more than 40 research projects and authored more than 100 scientific articles. From them, more than 40 are published in journals of the highest level. His research interests include artificial intelligence to intelligent transportation systems, including fuzzy-logic-based decisions, evolutionary optimization, and machine learning.

DIEGO OLIVA (Senior Member, IEEE) received the B.S. degree in electronics and computer engineering from the Industrial Technical Education Centre (CETI), Guadalajara, Mexico, in 2007, the M.Sc. degree in electronic engineering and computer sciences from Universidad de Guadalajara, Mexico, in 2010, and the Ph.D. degree in informatics from Universidad Complutense de Madrid, in 2015. He is currently an Associate Professor with Universidad de Guadalajara. His research interests include evolutionary and swarm algorithms, hybridization of evolutionary and swarm algorithms, and computational intelligence. He is a member of Mexican National Research System (SNII), the Association for Computing Machinery (ACM), and Mexican Academy of Computer Sciences (AMEXCOMP).

ANDER ARRIANDIAGA received the M.Sc. degree in control engineering, automatics and robotics from the University of the Basque Country, Bilbao, Spain, in 2011, and the Ph.D. degree in engineering from the Faculty of Engineering, University of the Basque Country. From 2016 to 2017, he was a Research Fellow with the University of the Basque Country, where he designed and developed deep neural networks for sequence classification in wire electrical discharge machining (WEDM). From 2018 to 2022, he was a Postdoctoral Researcher on dynamic vision sensors (DVS) and spiking neural networks with the Istituto Italiano di Tecnologia. Since 2022, he has been a Research Associate with DeustoTech, University of Deusto, where he applies machine learning techniques to control urban traffic. His research interests include machine learning, deep learning, and neuromorphic engineering.

ARKA GHOSH received the M.Tech. degree in information technology and the Ph.D. degree in computer science and technology from Indian Institute of Engineering Science and Technology Shibpur, Howrah, India, in 2015 and 2021, respectively. From 2020 to 2021, he was a Researcher with the ECRG Research Group, Victoria University of Wellington, on class imbalance learning and feature engineering. He is currently a Research Associate with the University of Deusto. He has published eight JCR papers and ten research papers in peer-reviewed international conference proceedings. His current research interests include soft computing, hybrid metaheuristics, machine learning, deep learning, and intelligent transportation systems.