A General Chip Subpixel Segmentation Localization Method Based on Improved Mayfly Algorithm

Weifeng Qian[1], Hao Sun[2,3*], Peng Shi[4,5,6], Imre Rudas[5]

[1]School of Electrical and Information Engineering, Northeast Petroleum University, Daqing, 163318, P.R. China; 2183314698@qq.com

[2]Bio-Computing Research Center, Harbin Institute of Technology, Shenzhen, Shenzhen; sunhao2021@hit.edu.cn

[3]China Shenzhen Medical Biometrics Perception and Analysis Engineering Laboratory, Harbin Institute of Technology, Shenzhen, Shenzhen, China; sunhao2021@hit.edu.cn

[4]School of Electrical and Mechanical Engineering, The University of Adelaide, SA 5005, Australia; peng.shi@adelaide.edu.au

[5]University Research and Innovation Centre, Óbuda University, Bécsi út 96/b, 1034 Budapest, Hungary

[6]School of Electronic, Electrical Engineering and Physics, Fujian University of Technology, Fujian, 350118, P.R. China

* Corresponding author, e-mail: sunhao2021@hit.edu.cn

Abstract: In order to improve the robustness of processing chip images in surface mount technology, especially when it comes to using a single threshold value under different lighting conditions, this paper aims to propose a chip localization algorithm with low computational complexity and high generality. The study investigates the multi-threshold-based online chip localization problem and introduces an intelligent optimization algorithm to enhance its performance. An automatic adjustment mayfly method is presented, improving the mayfly algorithm by combining it with the sine and cosine algorithm to enhance global search and convergence capabilities, resulting in improved fitness values. Additionally, image processing using inter-class variance yields multiple thresholds. Together with corner point detection, a versatile chip localization method is proposed. Simulation results demonstrate significant enhancements in solution accuracy, convergence speed, and merit-seeking capability achieved by the improved algorithm. Finally, the method's effectiveness is verified through various chip localization experiments.

Keywords: Subpixel; Multi-threshold segmentation; Mayfly algorithm; Chip localization

1 Introduction

Integrated circuit boards are the basis of electronic products and use the latest manufacturing method, surface mount technology [1]. Due to the rapid development of the electronics industry, the requirements for integrated circuit boards are gradually increasing [2]. To meet this demand, machine vision technology has been introduced into the surface mount process to improve the capability of the mounter in achieving high performance results. The mounter, as the key equipment of the surface mount process, needs to accomplish the task of inspection of surface mount chips with the help of machine vision technology. Machine vision technology enables functions such as learning of chip parameters, positioning, and defect detection [3-5]. Specifically, the inspection process of the placement machine includes the following steps: first, the PCB board is transferred to a fixed position through the motion track; then, the camera acquires the image of the component adsorbed by the suction nozzle; Next, register component types on the software interface and use image processing algorithms to teach and recognize components to obtain component parameters; finally, the component is accurately placed onto the PCB board. Through machine vision technology, the position and posture of the chip can be accurately detected and the chip positioning function can be realized, thus automating the placement process. Such technology application provides support for efficient and accurate surface mounting.

The first step in chip localization is segmentation of component images, and the commonly used methods for image segmentation [6-9] include threshold-based, edge-based, clustering-based and neural network-based methods [10]. One of the most frequently employed methods for chip image segmentation is the Otsu method [11, 12], a classical image segmentation algorithm that operates by dividing an image into two segments based on the distribution of its gray values. To calculate the position and direction of BGA, Ruo et al. [13] suggested a binary image-based localization approach utilizing the Rectangular Least Squares Rectangle approach. Xiao et al. [14] proposed the use of Otsu method and morphological filtering algorithm for various LED chip images, and then used template matching for chip localization. Amarulla Octavian et al. [15] proposed the combination mechanism of artificial intelligence and human intelligence in the swarm movement of defender drones to optimize the manoeuvre of defender drone swarms. Although the Otsu method is widely used in the field of image processing, it also has some drawbacks. For example, it is more sensitive to noise. In addition, the method only extracts the brighter regions of the chip, and the lighting condition of the image has a large impact on the results of chip information acquisition. If the image is too bright, i.e., the weight of high-value pixel points is larger, the image appears overall whitish. If the image is too dark, i.e., if there is a large proportion of low-value pixels, the image appears black. In the case of strong or weak illumination, the single-threshold segmentation algorithm cannot effectively segment the desired chip image, which makes the subsequent chip location acquisition algorithm difficult. In order to improve the segmentation effect, the single-threshold segmentation algorithm needs

to be extended to use multiple thresholds for segmentation, i.e., multi-threshold segmentation. In engineering applications, there are high requirements for component detection speed and accuracy. However, multi-threshold image segmentation algorithms suffer from low computational accuracy and long computation time. To mitigate these issues and improve the performance of multi-threshold algorithms, intelligent optimization algorithms are commonly employed to determine suitable multiple thresholds [16], thereby enhancing the overall operation of multi-threshold segmentation algorithms.

The mayfly algorithm [17] was originally proposed by Zervoudakis et al. in 2020, and combines the benefits of well-known optimization algorithms as Particle Swarm Optimization (PSO) [18], Genetic Algorithm (GA) [19] and Firefly Algorithm (FA) [20]. Traditional mathematical and numerical computational methods often face difficulties in solving complex functional optimization problems, especially for multi-objective optimization, non-convex analysis and problems with complex non-determinism [21]. As the search dimension increases and the search space grows exponentially, the traditional numerical solution methods become no longer applicable, so new algorithms are needed to solve these problems. And the mayfly algorithm has strong search and learning abilities, which can effectively solve single objective optimization problems. It applies the information exchange and cooperation between individuals to the optimization process by simulating the foraging behavior and social interaction behavior of mayflies. The individuals in the algorithm are regarded as mayflies and find the optimal solution by adjusting their position and speed. The Mayfly Optimization Algorithm exhibits superior global search and convergence capabilities, enabling it to uncover improved solutions within complex search spaces. It has demonstrated its versatility and applicability across various fields. For instance, Li et al. [22] successfully established a magnetic field compensation model using the mayfly algorithm to solve the three-axis magnetic configuration problem in the presence of disturbing magnetic fields. To address the global path planning issue for robots in complicated dynamic situations, Zou et al. [23] suggested an enhanced mayfly method based on Q learning. Liu et al. [24] applied an improved mayfly algorithm for estimating the optimal weight coefficients of sequence prediction values in wind speed prediction, and achieved accurate point prediction and interval prediction performance, which provides strong support for point planning and scheduling.

Both Shi-Tomasi and Harris are corner point detection methods based on gradient calculation. In comparison, Shi-Tomasi algorithm performs well in terms of stability, robustness to image rotation, lighting conditions, visual changes and noise, and inherits the advantages of Harris algorithm. In addition, the Shi-Tomasi algorithm has stronger adaptive ability to achieve uniform distribution of detected feature points and avoid clustering phenomenon.

Lighting conditions can affect imaging, causing image segmentation judgments and boundaries to be affected by conditions that are too dark or too bright [25]. In order to handle complicated chip kinds and satisfy the demands of high light change

stability, this paper proposes a general-purpose chip localization method. The method firstly introduces a dynamically adjusted mayfly algorithm and improves it by combining the sine and cosine factors to enhance the global search capability and convergence of the algorithm, so as to obtain better adaptation values. Secondly, in the placement machine project, the collected image is processed in real-time, and the segmentation algorithm is used to process the chip image. Finally, in combination with the Shi-Tomasi algorithm [26], multiple types of chips can be localized to obtain the chip location accurately. The overall process is shown in Figure 1. Through the above improvements combined with the Shi-Tomasi algorithm, the proposed method shows higher adaptability and stability, which can effectively meet the requirements of light change and complex chip positioning requirements.

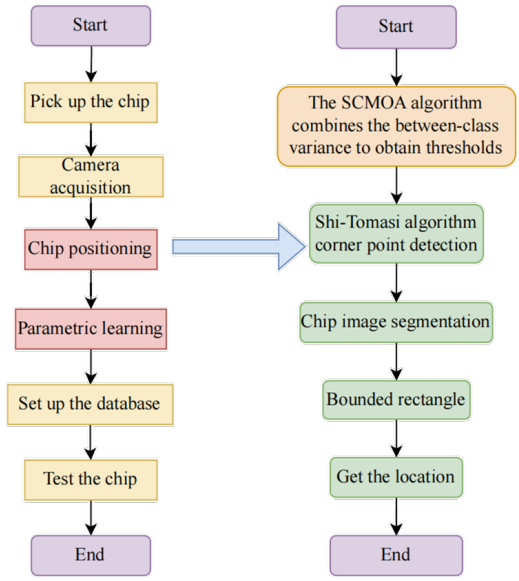

Figure 1
Flowchart of general chip positioning

2 Basic Mayfly Algorithm

An optimization algorithm called the mayfly optimization algorithm is based on the mating behavior of mayflies. In the algorithm, the position of individual mayflies is used to represent the solution of the algorithm. The mayfly algorithm simulates individual mayflies by creating a population of individuals. In this group, each individual represents a potential solution to a problem, and their position in the problem space reflects the characteristics of those solutions.

2.1 Male Mayfly Movement

During the courtship process of mayflies, males gather together in groups. By sensing the location and fitness of neighboring individuals, each individual male mayfly can adjust his position and speed to better find a mate or achieve other goals. This type of neighborhood interaction allows individual mayflies to act in concert by interacting with each other, thus enhancing the search and courtship effectiveness of the entire group. The formula is as follows:

$$x_{i,j}^{t+1} = x_{i,j}^{t} + v_{i,j}^{t+1} \tag{1}$$

where $x_{i,j}^{t+1}$ indicates the position of the $t+1$ individual in the i generation population in the j dimension. Through the neighborhood interaction and the update of individual positions in the mayfly optimization algorithm, the population of individuals is able to find a better solution in the problem space. This mayfly behavior-based optimization algorithm has strong search and learning capabilities.

Considering that mayflies are constantly moving and displaying dances on the surface of the water, we can adjust the way the velocity updates. Specifically, each individual mayfly's speed update will be influenced by other individuals within a certain distance around it. By interacting with neighboring individuals, individual mayflies can adjust their speed in order to perform a more coordinated dance on the surface of the water. This mechanism of neighborhood interaction allows individual mayflies to collaborate with each other in a group to achieve a more graceful dance. Thus the male mayfly speed equation can be expressed as:

$$v_{i,j}^{t+1} = \begin{cases} v_{i,j}^{t} + a_1 e^{-\beta r_p^{\,2}} \left(pbest_{i,j} - x_{i,j}^{t} \right) + \\ a_2 e^{-\beta r_g^{\,2}} \left(gbest_{i,j} - x_{i,j}^{t} \right) \\ v_{i,j}^{t} + d \cdot r \end{cases} \tag{2}$$

where $v_{i,j}^{t}$ denotes the velocity of the j dimension of the i individual in the population of the t generation, $pbest_{i,j}$ is the optimal position reached by that mayfly individual, $gbest_{i,j}$ is the global optimal position, a_1 and a_2 is the coefficient of attraction of the mayfly movement, β is the fixed visibility coefficient, a_1, a_2 and β are constants. r_p is the distance from the current position from $pbest_{i,j}$, r_g is the distance from the current position to $gbest_{i,j}$, d represents the dance coefficient, r is a random quantity between $[-1,1]$, and d the formula of

$$d^{t+1} = d^t \cdot ddamp \tag{3}$$

where d_t denotes the wedding dance coefficient at the time t and $ddamp$ is the dance damping.

With the above adjustments, the speed update of individual mayflies in the population will take into account the interaction with neighboring individuals, thus prompting them to act better together and achieve a more coordinated dance. This improved speed update further improves the effectiveness in problem solving.

2.2 Female Mayfly Movement

Although the behavioral patterns of female mayflies differ from those of males, their relative weakness in group aggregation does not prevent them from playing an important role in species reproduction. This behavioral difference between the sexes allows individual mayflies to play their respective roles in reproduction, thereby maintaining population stability and diversity. The formula is as follows:

$$v_{i,j}^{t+1} = \begin{cases} v_{i,j}^t + a_3 e^{-\beta r_{mj}^2} \left(x_{i,j}^t - y_{i,j}^t \right) \\ v_{i,j}^t + fl \cdot r \end{cases} \tag{4}$$

Where y_i^t represents the position of the female mayfly at the t iteration i, a_3 represents the positive attraction, r_{mf} is the distance between the female mayfly and the male mayfly, f_1 is the random wandering coefficient.

2.2 Mayfly Mating Process

In the mayfly algorithm, the mating process involves the selection of parents from among female and male individuals. The selection of parents is similar to the process by which male mayflies attract female mayflies, i.e., some mechanism is used to select individuals with a higher fitness level. Mating occurs between male and female individuals in the algorithm, where the optimal individual crosses with the optimal individual and the suboptimal individual crosses with the suboptimal individual. This mating process is carried out randomly, in which each pair of parents produces two offspring. By mating, the two offspring produced belong to the optimal and suboptimal individuals, respectively. This approach aims to use genetic crosses between individuals of higher fitness and those of lower fitness in the hope of obtaining better offspring. Through the mating process in the mayfly algorithm, genetic information is passed and exchanged between individuals, further improving the algorithm's search ability and optimization. The formula is shown in Equation (5):

$$\begin{aligned} offs1 &= L \cdot male + (1-L) \cdot female + \sigma N_1(0,1) \\ offs2 &= L \cdot female + (1-L) \cdot male + \sigma N_2(0,1) \end{aligned} \tag{5}$$

where $offs1$ and $offs2$ represents the male and female offspring, $male$ is the father, $female$ is the mother, L is a random number obeying Gaussian distribution between $[-1,1]$, $\sigma N_1(0,1)$ is a random number obeying Gaussian distribution within the mean value of 0 and variance of 1.

3 Improving the Mayfly Algorithm

3.1 Activation of Inertia Weights and Dynamic Adjustment of Mayfly Position

The inertia weighting mechanism in particle swarm algorithms is an important way of parameter tuning [27]. In the early iteration, larger inertial weights facilitate global search of particles for possible solutions. In the late iteration, the smaller inertial weights enable the particles to search more intensively for the local optimal solution, thereby accelerating the convergence process of the algorithm. Jinpeng Yu et al. [28] used neural networks to identify unknown nonlinear functions to improve the stability of stochastic closed-loop systems. Based on the above description, a nonlinear inertia weight factor (see Equation (6)) is proposed in this paper, inspired by the variation curve of the GLEU activation function. The introduction of this nonlinear amplitude factor can make the inertia weights have different variation characteristics during the iterative process.

$$w(t) = 0.5 \cdot \frac{t}{T}\left(1 + \tanh\left(\frac{t^3}{T^3}\right)\right) \tag{6}$$

where x_{mean}^{t} is the average of the total fitness of all individuals in the t generation. With the above adjustments, the optimal solution can be found faster by inertia weights at the beginning of the iteration, while the local optimum problem can be overcome at the later stage by switching between the original velocity and position update formulas to improve the search capability and optimization of the mayfly algorithm. This strategy can be better adapted to the characteristics of the problem. The location update of male mayflies uses the original update formula, and such a choice preserves the original update mechanism. Since the location update of male mayflies has an impact on the location update of female mayflies, by maintaining diversity in the location update of male mayflies, the exploration ability of the whole

population can be promoted, thus finding the optimal solution faster. This diversity of position update strategies provides a larger search space for the algorithm, increasing the flexibility and adaptability of the algorithm, helping to speed up the optimization process and improving the quality of the final results.

$$x_i^{t+1} = \begin{cases} wx_{ij}^t, \sigma \geq 0.001 \\ x_{ij}^t + v_i^{t+1}, \sigma < 0.001 \end{cases} \tag{7}$$

$$\sigma = \left| x_i^t - x_{\text{mean}}^t \right| \tag{8}$$

3.2 Crossover Variation

The sine cosine algorithm (SCA) [29] is proposed by the Australian scholar Seyedali Mirjalili in 2016. The algorithm is designed based on the sine cosine mathematical function and the algorithm starts with a set of random solutions, improves these random solutions by iteratively evaluating the objective function, and updates the position through a set of rules as the core of the optimization technique, and the position replacement is done through a linear decreasing function, i.e., the transformation parameter. The standard positional update formula of the sine and cosine algorithm is shown in equation (9). This algorithm improves the quality of optimization results by strengthening the combination of random solutions and focusing on the search space in a targeted manner.

$$x_{i,j}^{t+1} = \begin{cases} x_{i,j}^t + r_1 \sin r_2 \left| \ r_3 \ \text{pgbest}_i^t - x_{i,j}^t \right|, r_4 < 0.5 \\ x_{i,j}^t + r_1 \cos r_2 \left| \ r_3 \ \text{pgbest}_i^t - x_{i,j}^t \right|, r_4 \geq 0.5 \end{cases} \tag{9}$$

where $pgbest_i^t$ is the global extremum in the i dimension, t is the current iteration number, r_2 is the range [0,2π]of random numbers, r_3 is the range [-2,2]of random numbers, r_4 is the range [0,1]of random numbers, r_1 is the random number in the range, is called the transformation parameter and is a linearly decreasing function which decreases linearly to 0 as the number of repetitions rises, and its expression is:

$$r_1 = a - t\frac{a}{T} \tag{10}$$

where T is the maximum number of iterations, a is a constant and the value in this article is 2.

To balance the exploitation capability of the Mayfly algorithm, we replace the step factor in the basic positive cosine algorithm with a nonlinearly decreasing inertia weight factor. This makes it possible to better control how the algorithm searches.

In the mayfly algorithm, performing a mating operation when male and female 2 generations mayfly are in close proximity, or when male and female 2 generations are in a local optimum position, may result in premature convergence of the population to continue the search for a more optimal solution. This is known as premature convergence and can limits the efficiency of the algorithm. To be able to cope with the premature convergence phenomenon, in this article, we introduce the positive cosine factor when using the mayfly algorithm for mating and use the Euclidean distance to determine the premature convergence. We use the distance between individuals to make the judgment. When the mayfly population falls into a local optimum, the sine cosine algorithm is introduced and its oscillatory change property is used to act on the parent mayflies and the female mayflies. At the same time, we also introduce the inertia weighting factor mentioned before to induce the population to leave the local optimum behind solution and thus improve the global search capability of the mayfly algorithm. With such improvements, the mayfly algorithm can better jump out of the local best answer and improve the capability of global search when early convergence phenomenon is encountered, thus improving the performance and optimization results of the algorithm.

$$\begin{cases} offs1 = L \times male + (1-L) \times female + \sigma N_1(0,1) \\ offs2 = L \times female + (1-L) \times male + \sigma N_2(0,1) \\ offs1 = w \times x_i + r_1 \sin(r_2) |\ r_3\ \text{pgbestx}_i^t - x_i| + \sigma N_3(0,1) \\ offs2 = w \times y_i + r_1 \sin(r_2) |\ r_3\ \text{pgbesty}_i^t - y_i| + \sigma N_4(0,1) \end{cases} \tag{11}$$

$$d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{12}$$

d is the Mayfly male and female's Euclidean distance., set $d < 10^{-6}$ then, determine the population into local convergence, w is the inertia weight.

Based on the above improvements to MOA, Figure 2 displays the flow of the suggested algorithm.

4 Compare with other Intelligent Algorithms

To contrast the SCMOA algorithm's benefits and drawbacks proposed in this paper with other typical intelligent algorithms, GWO (Gray Wolf Optimization algorithm) [30], SCA (Sine Cosine Optimization algorithm), PSO (Particle Swarm

Optimization algorithm), the MAC improvement algorithm in the literature [31], and the standard mayfly algorithm are picked for contrast. These algorithms were run in the 12 standard test functions in Table 1. In the experiments, a setup with a dimension of 30, a total population size of 40, and a maximum number of 500 iterations was chosen. Each algorithm was run 30 times independently to

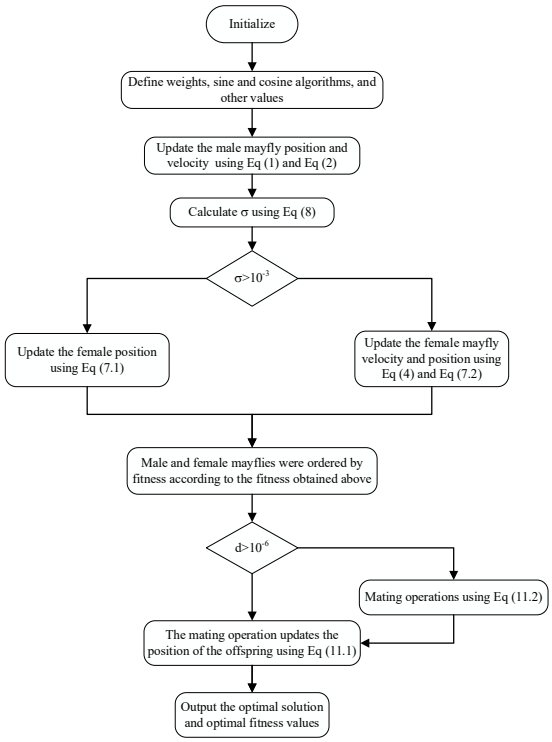

Figure 2

Flow chart of SCMOA algorithm

obtain stable results, and refer to Table 2 for information on specific algorithms' parameter settings. Using the aforementioned experimental configurations and comparisons, the performance of SCMOA algorithm and to assess their benefits and drawbacks and performance, other algorithms on various test functions can be compared. After effective quantitative analysis of these 12 test functions, the results as in Table 3 show that the SCMOA algorithm outperforms the original MOA.

Table 1

Benchmarking functions

Function	Name	Dimension	Range	Optimum value
f_1	Sphere	30	[-100，100]	0

f_2	Sehwefel’s2.22	30	[-10，10]	0
f_3	Sehwefel's2.21	30	[-100，100]	0
f_4	Quartic	30	[-1.28，1.28]	0
f_5	Power	30	[-100，100]	0
f_6	Schwefel's2.26	30	[-500,500]	-8379.66
f_7	Rastrigin's	30	[-5.12,5.12]	0
f_8	Ackley's	20	[-32,32]	0
f_9	Griewank's	30	[-600,600]	0
f_{10}	Penalized1	30	[-50,50]	0
f_{11}	Kowalik's	4	[-5,-5]	3.075E-04
f_{12}	Branin	2	[-5,10]	0.398

After effective quantitative analysis of these 12 test functions, Table 3's findings demonstrate that the SCMOA method performs better than the original MOA algorithm and the SCA algorithm in search on most of the functions. In seven functions, the SCMOA algorithm has reached the theoretical optimum and achieved better optimum values than other algorithms in other test functions. In particular f_{11}, the SCMOA algorithm has improved up to 5 orders of magnitude in the indefinite dimensional multi-peak function test function. This indicates that the SCMOA algorithm performs well in terms of solution search capability, accuracy and stability. After conducting 30 independent experiments, the mean values of the test functions f_1, f_6, f_8, and f_{10} are close to the theoretical optimum, indicating that the algorithm exhibits good stability. Furthermore, in all other test functions, the mean values of the SCMOA algorithm are superior to those of other algorithms. In addition, the standard deviation of SCMOA algorithm in most test functions is also smaller than other algorithms, which demonstrates that the algorithm has strong robustness. In summary, the SCMOA algorithm performs well in terms of search capability, accuracy and stability, and has better performance compared with other algorithms.

Table 2
Main parameter settings for different algorithms

Algorithm	Parameter Setting
SCMOA	$\alpha = 1.0,\ f1 = 1, fldamp = 0.99, d = 5. ddamp = 0.8$ $a_1 = 1.0, a_2 = 1.5, a_3 = 1.5, \beta = 2, g_0 = 0.8$

	$a_1 = a_2 = 1.5$
SCA	A=2
	$a = [2,0]$ Linear decreasing
MOA	Same as SCMOA
MAC	Same as SCMOA

Table 3
Comparison of benchmark functions

Functions	$f_{\min}$		SCMOA	PSO	SCA	GWO	MOA	MAC
f_1	0	avg	00E+00	7.709E-01	8.321E+00	5.422E-31	4.877E-08	00E+00
		best	00E+00	3.743E-01	1.621E-03	2.206E-32	5.742E-12	00E+00
		std	00E+00	2.188E-01	2.046E+01	9.065E-31	1.746E-07	00E+00
f_2	0	avg	8.40E-323	5.130E+00	1.150E-02	1.635E-18	6.551E-04	5.08E-264
		best	00E+00	2.751E+00	3.789E-04	2.368E-19	5.147E-07	1.77E-287
		std	00E+00	1.656E+00	1.854E-02	1.044E-18	2.816E-03	00E+00
f_3	0	avg	9.88E-324	2.595E+00	3.153E+01	9.181E-08	7.956E-01	3.21E-270
		best	00E+00	1.127E+00	1.346E+01	4.149E-09	1.953E-01	1.39E-293
		std	00E+00	9.363E-01	1.359E+01	9.363E-01	5.027E-01	00E+00
f_4	0	avg	1.523E-04	3.775E+00	7.687E-02	1.624E-03	1.469E-02	1.313E-04
		best	3.360E-07	7.106E-02	1.276E-02	5.780E-04	6.754E-03	2.347E-06
		std	1.604E-04	7.273E+00	9.057E-02	7.174E-04	5.007E-03	9.496E-05
f_5	0	avg	00E+00	1.776E-04	3.306E-05	4.34E-105	2.009E-28	00E+00
		best	00E+00	2.686E-06	1.817E-11	3.12E-116	3.197E-38	00E+00
		std	00E+00	1.422E-04	7.469E-05	2,17E-104	6.320E-28	00E+00
f_6	8379.66	avg	8.204E+03	3.443E+03	3.715E+03	5.899E+03	8.466E+03	1.256E+04
		best	6.726E+03	5.949E+03	4.295E+03	7.045E+03	6.135E+03	1.257E+04
		std	6.932E+02	7.226E+02	2.980E+02	8.96E+02	7.250E+02	7.776E+00
f_7	0	avg	00E+00	9.358E+00	1.669E+00	4.907E-01	1.239E+00	00E+00
		best	00E+00	4.345E+00	00E+00	00E+00	00E+00	00E+00
		std	00E+00	3.407E+00	4.292E+00	1.289E+00	1.068E+00	00E+00
f_8	0	avg	8.882E-16	3.049E+00	1.495E+01	5.690E-14	3.474E+00	8.882E-16
		best	8.882E-16	1.870E+00	4.657E-02	3.997E-14	1.647E+00	8.882E-16
		std	00E+00	6.124E-01	9.387E-01	8.586E-15	9.387E-01	00E+00
f_9	0	avg	00E+00	3.502E-01	7.712E-01	5.568E-03	2.672E-02	00E+00
		best	00E+00	1.298E-01	8.505E-03	00E+00	1.548E-09	00E+00
		std	00E+00	1.130E-01	2.890E-01	7.739E-03	1.680E-02	00E+00
f_{10}	0	avg	1.821E-09	3.567E+00	7.258E+04	3.317E-02	4.287E-01	3.217E-03
		best	7.991E-14	8.488E-01	6.451E-01	6.372E-03	1.497E-09	2.333E-04
		std	4.374E-09	1.244E+00	3.445E+05	1.922E-02	5.015E-01	2.791E-03
f_{11}	3.075E-04	avg	3.075E-04	3.846E-03	9.531E-04	8.911E-03	1.007E-03	3.234E-04
		best	3.075E-04	4.216E-04	3.679E-04	3.075E-04	3.075E-04	3.094E-04
		std	2.925E-19	6.739E-03	3.435E-04	1.308E-02	3.659E-03	1.286E-05
f_{12}	0.398	avg	3.979E-01	3.979E-01	3.996E-01	3.979E-01	3.979E-01	4.007E-01
		best	3.979E-01	3.979E-01	3.979E-01	3.979E-01	3.979E-01	3.979E-01
		std	00E+00	7.883E-05	1.907E-03	1.169E-06	00E+00	00E+00

5 Practical Engineering Applications

5.1 Chip Positioning

For the image with strong illumination, as shown in Figure 3 below, we obtained a threshold value of 189 using the improved mayfly algorithm combined with multi-threshold segmentation, while the threshold value obtained using the Otsu method is 130. The figure below shows the binarized image obtained by segmenting the chip image according to these thresholds. It is clearly seen that the threshold segmentation obtained using the improved method is better.

Figure 3
The left side is the original image, the center is the Otsu method segmentation image, and the right side is the improved algorithm to obtain the threshold segmentation image

When processing chip images containing elements such as pins, nozzles, chip body and background, we set the K value to 4 to obtain four thresholds in an adaptive manner. By sorting these four thresholds, we select the pixel between the last two thresholds as the pixel of the chip pin part. By binarizing the chip image, we can detect the corner locations of the chip using the Shi-Tomasi algorithm, an image feature-based corner detection algorithm that finds corner points in the image and calculates their quality scores. By applying this algorithm, we are able to accurately locate the corner point position of the chip, which provides important information for the subsequent localization and processing steps.

Next, we can use the calculated feature points to calculate the minimum outer rectangle and subsequently discover the four vertices' coordinates for the smallest outer rectangle. With these vertex coordinates, we are able to accurately locate the position of the chip. This method is applicable to many types of chip localization, such as SOJ chips, QFP chips, TR chips, and irregular chips.

5.2 Stability Test

We contrast the proposed approach in this paper with the method from the literature to ensure the efficacy of the proposed identification and localization method [32]. We collected 256 images of these four chips under different illumination and angles, which in total constitute a dataset containing 1024 images.

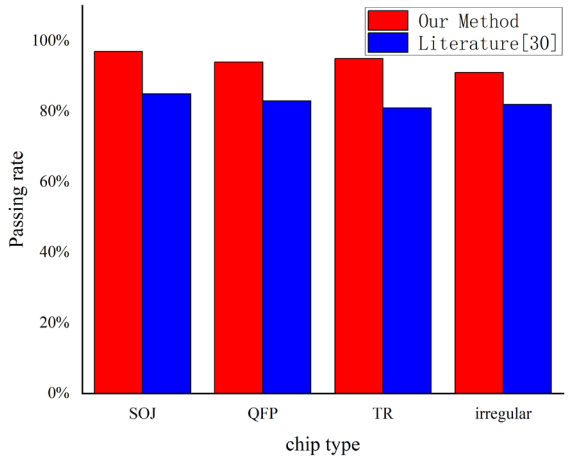

Figure 4
Pass rates for four types of chips

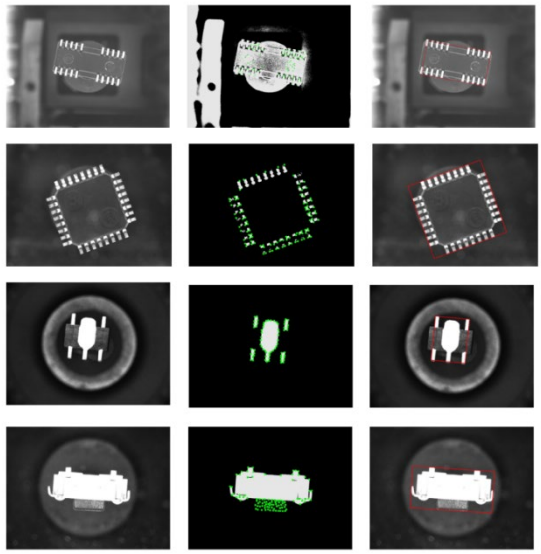

Figure 5
Results based on several chips. The left image in each set of chip images is the original, the center image displays the boundary feature points that were extracted for the binary image, and the right image displays the chip's smallest closed rectangle

We contrasted the recognition rate that is, the proportion of correctly found photos to all images between groups of images. The experimental results, shown in Figure 4, show that our method consistently has a recognition rate of more than 92% and is largely independent of chip type, being able to locate chips with light interference

and irregular shapes. Figure 5 lists four types of chip binary recognition and positioning.

5.3 Precision Test

Figure 5 is an image with a resolution of 1288 pixels * 964 pixels, use the drawing tool to read the pixel coordinates of the chip's corner points as the actual coordinates compare this paper's algorithm with the template matching algorithm, and compare the detection results of its four types of components in the positions shown in the figure, and the characteristic parameters obtained are shown in Table 4. From Table 4, it can be seen that the algorithm in the paper is better than template matching, because the proposed multi-threshold method is used to pre-process the chip to ensure that the characteristics of the chip, the maximum error of the chip center coordinates under the algorithm is 0.8 pixels, which meets the requirements of high precision and high quality.

Table 4
Comparison of Coordinates and Errors

Element	Standard Coordinate	Template Match Coordinate		Algorithm Coordinate	
		Calculate coordinates	Error	Calculate coordinates	Error
SOJ	(663,438)	(663.764,438.733)	1.059	(663.336,438.117)	0.355
QFP	(657,506)	(657.223,506.412)	0.468	(657.118,506.235)	0.263
TR	(636,436)	(636.647,436.764)	1.001	(636.133,436.746)	0.758
Irregular	(655,480)	(655.705,480.941)	1.176	(655.199,480.602)	0.634

Conclusions

This paper introduces a universal chip localization method that combines an improved mayfly algorithm and multi-threshold segmentation. Its purpose is to meet the stringent stability requirements posed by changing lighting conditions and complex chip types. By dynamically adjusting the mayfly algorithm and enhancing the search strategy, we achieve improved convergence and solution quality across 12 different test functions. With the Shi-Tomasi algorithm, we successfully achieve stable and accurate localization of various chip types, even in varying lighting conditions. In experiments involving 1024 chip images with diverse lighting and angles, our method achieves a pass rate of over 92%, irrespective of chip types. Compared to existing methods, ours is more robust, offering a practical solution to chip localization with broad application potential.

Acknowledgement

The work presented in this paper was supported by Natural Science Foundation of China (Grant No. 62203141), and the Australian Research Council (DP240101140).

References

[1] Acciani, G., Brunetti, G., & Fornarelli, G. Application of neural networks in optical inspection and classification of solder joints in surface mount technology. IEEE Transactions on Industrial Informatics, (2006) 2(3), 200-209

[2] Illés, B., Piller, I., Radeleczki, S., Tóth, T., & Wagner, G. (2023) A Novel Approach of Operation Sequencing Problem in Computer Aided Process Planning. Acta Polytechnica Hungarica, 20(6), 173-193

[3] Liu, W., Yang, X., Sun, H., Yang, X., Yu, X., & Gao, H. (2022) A Novel Subpixel Circle Detection Method Based on the Blurred Edge Model. IEEE Transactions on Instrumentation and Measurement, 71, 1-11

[4] Bai, L., Yang, X., & Gao, H. (2018) Corner Point-Based Coarse–Fine Method for Surface-Mount Component Positioning. IEEE Transactions on Industrial Informatics, 14(3), 877-886

[5] Liu, W., Yang, X., Yang, X., & Gao, H. (2022) A novel industrial chip parameters identification method based on cascaded region segmentation for surface mount equipment. IEEE Transactions on Industrial Electronics, 69(5), 5247-5256

[6] Jianhua, Z., Qiang, Z., Jinrong, Z., Lin, S., & Jilong, W. (2019) A novel algorithm for threshold image denoising based on wavelet construction. Cluster Computing, 22(S5), 12443-12450

[7] Liu, G., & Zhou, H. (2013) Semantic segmentation based on multi-stage region-level clustering. SPIE Proceedings, MIPPR 2013: Multispectral Image Acquisition, Processing, and Analysis. Presented at the Eighth International Symposium on Multispectral Image Processing and Pattern Recognition, Wuhan, China

[8] Zhang, Z., Fan, X., Xie, Y., & Xu, H. (2018) An edge detection method based artificial bee colony for underwater dam crack image. Biomedical Imaging and Sensing Conference

[9] Gavrilov, D. A. (2021) Investigation of the applicability of the convolutional neural network U-Net to a problem of segmentation of aircraft images. Computer Optics, 45(4)

[10] Zhang, H., Zhao, N., Wang, S., & Agarwal, R. K. (2023) Improved Event-Triggered Dynamic Output Feedback Control for Networked T–S Fuzzy Systems With Actuator Failure and Deception Attacks. IEEE Transactions on Cybernetics, 53(12), 7989-7999

[11] Kwon, S.-H., Son, S.-H., & Bae, J.-I. (2003) Threshold Selection Method Based on the Distribution of Gray Levels. Journal of Korean Institute of Intelligent Systems, 13(6), 649-654

[12] Afifah A. N. N., Indrabayu, Suyuti, A., and Syafaruddin (2023), Improving the Image Quality of Grayscale Thermal Images Taking from Photovoltaic Panel with Contrast Enhancement Method, International Journal of Innovative Computing, Information and Control, 19(1), pp. 197-212, https://doi.org/10.24507/ijicic.19.01.197

[13] Shih, C.-L., Ruo, C.-W., & Sheu, H.-T. (2005) Locating and checking of BGA pins' position using gray level. The International Journal of Advanced Manufacturing Technology, 26(5-6), 491-498

[14] Xiao, W. (2013) Research of LED Chips Positioning System. Advanced Materials Research, 774-776, 1512-1517

[15] Amarulla Octavian, Wisnu Jatmiko, Ario Yudo Husodo and Grafika Jati (2023) Optimization of Defender Drone Swarm Battle Maneuver for Gaining Air Superiority by Combining Artificial and Human Intelligence through Hand Gesture Control System. International Journal of Innovative Computing, Information and Control, 19(2): 623-636.

[16] Du, M., Ding, Y., & Jia, Q. (2013) A multi-threshold segmentation method based on ant colony algorithm. SPIE Proceedings, Fifth International Conference on Machine Vision (ICMV 2012): Algorithms, Pattern Recognition, and Basic Technologies. Presented at the Fifth International Conference on Machine Vision (ICMV 12), Wuhan, China

[17] Zervoudakis, K., & Tsafarakis, S. (2020) A mayfly optimization algorithm. Computers & Industrial Engineering, 106559

[18] Poli, R., Kennedy, J., & Blackwell, T. (2007) Particle swarm optimization. Swarm Intelligence, 33-57

[19] Lambora, A., Gupta, K., & Chopra, K. (2019) Genetic Algorithm- A Literature Review. 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon). Presented at the 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India

[20] Yang, X.-S. (2009) Firefly algorithms for multimodal optimization. In Stochastic Algorithms: Foundations and Applications,Lecture Notes in Computer Science (pp. 169-178)

[21] Habashneh, M., Rad, M. M., & Fischer, S. (2023) Bi-directional Evolutionary, Reliability-based, Geometrically Nonlinear, Elasto-Plastic Topology Optimization, of 3D Structures. Acta Polytechnica Hungarica, 20(1), 169-186

[22] LI, L., LIU, W., & LI, L. (2022) Underwater magnetic field measurement error compensation based on improved mayfly algorithm. Xibei Gongye Daxue Xuebao/Journal of Northwestern Polytechnical University, 40(5), 1004-1011

[23] Zou, A., Wang, L., Li, W., Cai, J., Wang, H., & Tan, T. (2023) Mobile robot path planning using improved mayfly optimization algorithm and dynamic window approach. The Journal of Supercomputing, 79(8), 8340-8367

[24] Liu, Z., Jiang, P., Wang, J., & Zhang, L. (2021) Ensemble forecasting system for short-term wind speed forecasting based on optimal sub-model selection and multi-objective version of mayfly optimization algorithm. Expert Systems with Applications, 177, 114974

[25] H. Zhang, H. Sun, W. Ao, and G. Dimirovski (2021) A Survey on Instance Segmentation: Recent Advances and Challenges. International Journal of Innovative Computing, Information and Control, 17(3), 1041-1053

[26] Jianbo Shi & Tomasi. (1994) Good features to track. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition CVPR-94. Presented at the Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA

[27] Ilham, A. A., Warni, E., and Pahrul (2024) Soft Voting Classifier with Optimized Weight Using Particle Swarm Optimization on Sentiment Analysis for Online Credit and Loan Application Reviews, International Journal of Innovative Computing, Information and Control, 20(2), pp. 359-372, https://doi.org/10.24507/ijicic.20.02.359

[28] Yu, J., Cheng, S., Shi, P., & Lin, C. (2023) Command-Filtered Neuroadaptive Output-Feedback Control for Stochastic Nonlinear Systems With Input Constraint. IEEE Transactions on Cybernetics, 2301-2310

[29] Mirjalili, S. (2016) SCA: A Sine Cosine Algorithm for solving optimization problems. Knowledge-Based Systems, 120-133

[30] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014) Grey wolf optimizer. Advances in engineering software, 69, 46-61

[31] Zhang, J.-H., Gao, Z.-M., Li, S.-R., & Zhao, J. (2022) Improved Mayfly Optimization Algorithm with Cooperation. 2022 7th International Conference on Computer and Communication Systems (ICCCS). Presented at the 2022 7th International Conference on Computer and Communication Systems (ICCCS), Wuhan, China

[32] Han, B., & Yi, M. (2018) A Template Matching Based Method for Surface-Mount Rectangular-Pin-Chip Positioning and Defect Detection. 2018 Eighth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC) Presented at the 2018 Eighth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC) Harbin, China