

## Article

# Combining Improved Meanshift and Adaptive Shi-Tomasi Algorithms for a Photovoltaic Panel Segmentation Strategy

Chao Huang <sup>1,2</sup>, Xuewei Chao <sup>1,2,\*</sup>, Weiji Zhou <sup>1,2</sup> and Lijiao Gong <sup>1,2</sup>

<sup>1</sup> College of Mechanical and Electrical Engineering, Shihezi University, Shihezi 832000, China;

20212009008@stu.shzu.edu.cn (C.H.); zhouweiji@stu.shzu.edu.cn (W.Z.); gonglijiao@stu.shzu.edu.cn (L.G.)

<sup>2</sup> Bingtuan Energy Development Institute, Shihezi University, Shihezi 832000, China

\* Correspondence: chaoxwemil@163.com

**Abstract:** To achieve effective and accurate segmentation of photovoltaic panels in various working contexts, this paper proposes a comprehensive image segmentation strategy that integrates an improved Meanshift algorithm and an adaptive Shi-Tomasi algorithm. This approach effectively addresses the challenge of low precision in segmenting target regions and boundary contours in routine photovoltaic panel inspection. Firstly, based on the image information of photovoltaic panels collected under different environments by cameras, an improved Meanshift algorithm based on platform histogram optimization is used for preliminary processing, and images containing target information are cut out; then, the adaptive Shi-Tomasi algorithm is used to extract and screen feature points from the target area; finally, the extracted feature points generate the segmentation contour of the target photovoltaic panel, achieving accurate segmentation of the target area and boundary contour of the photovoltaic panel. Experiments verified that in photovoltaic panel images under different background environments, the method proposed in this paper enhances the accuracy of segmenting the target area and boundary contour of photovoltaic panels.

**Keywords:** photovoltaic panel; image segmentation; Meanshift algorithm; Shi-Tomasi algorithm



**Citation:** Huang, C.; Chao, X.; Zhou, W.; Gong, L. Combining Improved Meanshift and Adaptive Shi-Tomasi Algorithms for a Photovoltaic Panel Segmentation Strategy. *Processes* **2024**, *12*, 564. <https://doi.org/10.3390/pr12030564>

Academic Editors: Pedro Dinis Gaspar, Pedro Dinho da Silva and Luís C. Pires

Received: 30 January 2024

Revised: 1 March 2024

Accepted: 8 March 2024

Published: 13 March 2024



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

## 1. Introduction

Energy is an important pillar of economic and social development, and it is also the main source of carbon emissions. Currently, the world's energy development is moving towards a new era of cleanliness, low carbonization, and intelligence. The use of clean energy sources such as photovoltaic, wind, nuclear, and hydroelectric power to replace fossil fuel-based power generation can effectively promote energy cleanliness. Among these, photovoltaic power generation has significant potential due to its cost advantages, and its penetration rate is expected to continuously rise, gradually becoming a primary energy source. Currently, with the rapid development of the photovoltaic industry, the photovoltaic sector is experiencing a trend of continuously increasing demand, accelerated expansion of installed capacity, and a more diverse range of application scenarios. This has raised higher requirements and more challenges for the operational management of photovoltaic power stations, which largely depend on the condition of the photovoltaic panels [1]. In recent years, image segmentation technology has been widely applied in both military and civilian sectors. It has also been adopted for the operational monitoring of photovoltaic power stations. By leveraging image segmentation techniques, substantial labor detection costs can be saved, and the efficiency of inspection can be significantly enhanced. This holds significant implications for the daily maintenance and operation of photovoltaic power stations [2,3]. However, as the diversification of photovoltaic panel application environments becomes the primary trend in the future, current image segmentation techniques face challenges in accurately segmenting target areas and boundary contours in the diverse photovoltaic panel application environments. This paper will primarily address and investigate these two issues.

To effectively segment photovoltaic panels, considering the characteristics of photovoltaic panel images, scholars, both domestically and internationally, have proposed numerous segmentation methods. These can be broadly categorized into the following: threshold-based segmentation methods [4], edge detection-based segmentation methods [5], region-based segmentation methods [6], and machine learning-based methods [7–10]. The threshold-based segmentation method has good results in images with good grayscale differences. Among threshold segmentation methods, the Otsu threshold method is a relatively classic segmentation method. In reference [11], an improved two-dimensional Otsu algorithm and a black widow spider optimization algorithm are proposed. This algorithm can calculate the optimal threshold with modest computational complexity, effectively enhancing the speed of threshold search and the efficiency of image segmentation. However, threshold-based segmentation algorithms obtain segmentation results by solving the gray-level frequency information contained in an image, resulting in poor segmentation accuracy when the background gray level is similar to that of the photovoltaic panel. Reference [12] presents a segmentation method based on K-means clustering, primarily utilizing the elbow method (EM) and gap methods to cluster thermal images into multiple regions and identify damaged areas. However, this approach does not account for the similar spatial information surrounding photovoltaic panel images, leading to reduced segmentation accuracy when there is similar spatial structure information present in the photovoltaic panel background environment. Reference [13] introduces the segmentation of photovoltaic panel images by combining standard image processing procedures with the edge detection operator (Canny), enhancing the speed and precision of the segmentation process. However, this approach only applies when no similar contours exist in the background. The method for segmenting photovoltaic panels based on infrared images as described in reference [14] is discussed. However, due to natural environmental disturbances, atmospheric attenuation, and inherent limitations of detectors, the spatial resolution of infrared images is generally low, leading to phenomena such as detail loss, blurred edges, and pronounced noise. Furthermore, in visible light images, both the background and target areas become more intricate, resulting in a more indistinct outline segmented by this algorithm. Consequently, it cannot effectively adapt to the segmentation of photovoltaic panels in complex environments. Reference [15] visually constructs a state-of-the-art semantic segmentation algorithm framework using tables to enumerate methods and data under various categories. Semantic segmentation based on convolutional neural networks not only identifies objects present in the image but also assigns a semantic label to each pixel. They can automatically learn features in images, helping computers achieve a more detailed and accurate understanding of objects in images, thereby greatly improving the accuracy of algorithms. However, due to the backpropagation characteristics of convolutional neural networks, semantic segmentation based on convolutional neural networks requires a large amount of data to support them, and the existence of pooling layers may lead to the loss of valuable information, resulting in low accuracy in edge segmentation under small sample conditions.

Based on the research and analysis of the aforementioned methods, it is evident that there are limitations and challenges in overcoming them in practical applications. To address the challenges of poor target area segmentation accuracy and low precision in segmentation boundary contours in various photovoltaic panel application environments, this paper proposes an image segmentation strategy based on an enhanced fusion of Mean-shift clustering and the adaptive Shi-Tomisa algorithm. This integrated approach involves preliminary image clustering and segmentation followed by the precise extraction of feature points from the target region, thereby generating a comprehensive segmentation contour. This strategy aims to enhance the segmentation performance of photovoltaic panels in environments with interference elements. Moreover, the segmentation contour generated based on feature points exhibits notable clarity and accuracy. The enhanced Meanshift algorithm presented in this paper addresses the initial value sensitivity of traditional Mean-shift clustering. Building upon the original algorithm, this study leverages the platform

histogram equalization algorithm to optimize the image histogram grayscale for image enhancement. Subsequently, the initial clustering center is determined based on a peak calculation formula to refine the original Meanshift algorithm. Moreover, an adaptive Shi-Tomisa algorithm is developed for feature point extraction. Through image segmentation experiments in several typical environments, it is proven that the segmentation strategy presented in this paper can effectively remove regions in the image information that are unrelated to the target object. Direct feature-point-based contour segmentation and generation are performed on the target area of the photovoltaic panel, achieving improved accuracy in segmenting the target area and boundary contour of the photovoltaic panel in different environments.

## 2. Materials and Methods

### 2.1. Improved Meanshift Algorithm

#### 2.1.1. Core Principles of the Meanshift Algorithm

The Meanshift algorithm is widely applied in clustering, image smoothing, segmentation, tracking, and various other domains. In traditional machine learning clustering algorithms such as K-Means, the initial clustering centers influence the final clustering outcome. The introduction of the K-Means++ algorithm provides a basis for selecting better initial clustering centers. However, in these algorithms, the number of clusters  $k$  still needs to be predetermined. For datasets where the number of clusters is not known in advance, both K-Means and K-Means++ pose challenges for accurate solutions. To address this, several improved algorithms have been proposed to handle scenarios where the number of clusters is unknown. Like K-Means, the Meanshift algorithm is a clustering algorithm based on cluster centers. However, unlike K-Means, the Meanshift algorithm does not require the number of clusters to be predetermined. The core operation of the Meanshift algorithm involves calculating the drift vector of the center point through data density changes within the region of interest. This drift vector moves the center point for the next iteration until it reaches the region with the maximum density (with the center point unchanged). This process can be applied starting from each data point. During this iteration, the number of data points within the region of interest is counted, and this parameter serves as the basis for classification in the end. Essentially, the Meanshift algorithm is an iterative process that uses parametric density estimation to identify local extrema within a dataset's density distribution [16]. The specific derivation is as follows:

For a given set of  $n$  sample points,  $i = 1, \dots, n$ , in a  $d$ -dimensional space  $R^d$ , select any point  $x$  within the space. The basic form of the Meanshift vector is defined as follows:

$$M_h = \frac{1}{K} \sum_{x_i \in s_h} (x_i - x) \quad (1)$$

In Equation (1),  $K$  denotes the number of points within the  $n$  sample points  $x_i$  that fall within the  $s_h$  region.  $S_h$  is a set of points  $y$  within a high-dimensional spherical region with a radius of  $h$ , satisfying the following relationship:

$$s_h(x) = \left\{ y : (y - x_i)_T (y - x_i) < h^2 \right\} \quad (2)$$

In the  $d$ -dimensional space, select any point and, using this point as the center, construct a hypersphere with radius  $h$ . Since there are  $d$  dimensions,  $d$  of which may be greater than 2, it is a high-dimensional sphere. All points within this sphere and the center will produce a vector, which is the endpoint of the vector starting from the center and ending at the point within the sphere. Then, add all these vectors together. The result is the Meanshift vector. Take the endpoint of the Meanshift vector as the center and construct another hypersphere. By repeating these steps, one can obtain a Meanshift vector. By repeating this process, the Meanshift algorithm can converge to the place where the probability density is maximum, that is, the most densely populated place.

### 2.1.2. Meanshift Algorithm with Kernel Function

Regarding the fundamental Meanshift formulation described in Equations (1) and (2), there exists an issue: within the region of  $S_h$ , every point contributes equally to  $x$  [17]. However, in reality, this contribution is related to the distance from  $x$  to each point. Moreover, the significance of each sample varies. Based on these considerations, a kernel function and sample weights are incorporated into the basic Meanshift vector formulation, resulting in the following improved Vector Meanshift formulation:

$$M_h(x) = \frac{\sum_{i=1}^n G_H(x_i - x) w(x_i)(x_i - x)}{\sum_{i=1}^n G_H(x_i - x) w(x_i)} \quad (3)$$

In Equation (3),

$$G_H(x_i - x) = |H|^{-\frac{1}{2}} G(H^{-\frac{1}{2}}(x_i - x)) \quad (4)$$

$G(x)$  is a unit kernel function.  $H$  is a positive-definite symmetric  $d \times d$  matrix, referred to as the bandwidth matrix, which is a diagonal matrix.  $w(x_i) \geq 0$  represents the weight of each sample. The diagonal form of  $H$  is as follows:

$$H = \begin{pmatrix} h_1^2 & 0 & \cdots & 0 \\ 0 & h_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_d^2 \end{pmatrix}_{d \times d} \quad (5)$$

The above vector (3) can be rewritten as:

$$M_h(x) = \frac{\sum_{i=1}^n G\left(\frac{x_i - x}{h_i}\right) w(x_i)(x_i - x)}{\sum_{i=1}^n G\left(\frac{x_i - x}{h_i}\right) w(x_i)} \quad (6)$$

For a given set of  $n$  sample points,  $i = 1, \dots, n$ , in a  $d$ -dimensional space  $R^d$ , the multivariate kernel density estimate at point  $x$  is as follows:

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^n k\left(\left\|\frac{x - x_i}{h}\right\|^2\right) \quad (7)$$

In the above equation,  $k(x)$  is the side profile function and  $h$  is the window radius. Let the gradient of Equation (7) be 0. The stagnation point of the density function is obtained as:

$$x = \frac{\sum_{i=1}^n x_i \cdot g\left(\left\|\frac{x - x_i}{h}\right\|^2\right)}{\sum_{i=1}^n g\left(\left\|\frac{x - x_i}{h}\right\|^2\right)} \quad (8)$$

In Equation (8),  $g(x) = -k'(x)$  serves as the side profile function of the kernel function  $G(x)$ . By continuously iterating to calculate the Meanshift vector and shifting until the program converges, one can locate the local mode. The iterative equation is as follows:

$$y_{j+1} = \frac{\sum_{i=1}^n x_i g\left(\left\|\frac{y_j - x_i}{h}\right\|^2\right)}{\sum_{i=1}^n g\left(\left\|\frac{y_j - x_i}{h}\right\|^2\right)}, j = 1, 2, \dots, \quad (9)$$

In Equation (9),  $y$  is the initial position center of the kernel window;  $y_{j+1}$  is the weighted average calculated using kernel  $G$  and window radius  $h$  at  $y_j$ .

For the Epanechnikov kernel function:

$$k_E(x) = \begin{cases} (2C_d)^{-1}(d+2)(1-x^T x) & , \text{ if } x^T x < 1 \\ 0 & , \text{ otherwise} \end{cases} \quad (10)$$

In the Equation (10),  $C_d$  represents the volume of a  $d$ -dimensional sphere.

Using the Meanshift algorithm with the kernel function, this algorithm can explore potential probability density modes. The algorithm can be described through the following steps:

- (1) Divide the feature space into  $n$  non-overlapping, equally sized partitions. The size of each partition is  $(G_{\max} - G_{\min})/n$ , where  $G_{\max}$  and  $G_{\min}$  represent the maximum and minimum grayscale values of image pixels. To avoid low-density regions, it is necessary to ensure that the number of pixels in each section is not less than a threshold value  $T_1$ .
- (2) Run the Meanshift program  $n$  times to obtain  $n$  convergent points. The Meanshift program employs a kernel radius of  $h = (G_{\max} - G_{\min})/n$ .
- (3) Merge adjacent convergence points that are closer than a preset threshold  $T_2$  into a single point and determine the potential center of  $m$  ( $m < n$ ) probability density modes.

#### 2.1.3. Utilizing Optimized Histogram for Initial Cluster Centers

The key operation of the Meanshift algorithm is to calculate the displacement vector of the center point through the data density change within the region of interest, thereby moving the center point for the next iteration until it reaches the maximum density (with the center point unchanged). From this, it can be inferred that the selection of the initial center point has a significant impact on the classification performance. Based on the characteristics of photovoltaic panel images, it can be roughly divided into the following two regions: the photovoltaic panel and the background area. However, in complex environments, the gray-level histograms of the background and panel areas often have multiple peaks and valleys, and even multiple closely spaced peaks. To effectively distinguish the gray-level histogram of the photovoltaic panel from the complex background, it is necessary to enhance the original image. Histogram equalization is a commonly used image enhancement method based on images [18]. It adjusts gray levels according to the cumulative histogram of the image to enhance the image. The gray level adjustment strategy is as follows: in the histogram, the intervals between pixels with many and densely distributed gray levels become larger, enhancing the contrast; for pixels with few and sparsely distributed gray levels, the intervals become smaller, even zero (the gray levels are merged), to reduce contrast. If general histogram equalization is used to enhance infrared images, it will result in more gray levels occupied by background and noise, and fewer gray levels for targets, which is equivalent to increasing the contrast of background and noise and reducing the contrast of targets. Therefore, to overcome the shortcomings of ordinary histogram equalization algorithms, this paper uses a platform histogram equalization algorithm to optimize images.

The platform histogram is a modification of the histogram. It adjusts the statistical histogram as follows by selecting an appropriate platform threshold  $T$ :

$$P_T(k) = \begin{cases} P(k), P(k) \leq T \\ T, P(k) > T \end{cases} \quad (11)$$

where  $k$  represents the grayscale level of the image ( $0 \leq k \leq 255$ ).  $P_T(k)$  is the platform histogram of the image, and  $P(k)$  is the statistical histogram of the image.  $T$  is the platform threshold. As observed in Equation (11), when  $T \rightarrow \infty$ , for  $k \in [0, 255]$ ,  $P_T(k) = P(k)$ , indicating that the platform histogram transforms into the statistical histogram. Therefore, the statistical histogram is a special form of the platform histogram. The equalization process for platform histograms is similar to that for histogram equalization, with the

following few exceptions: in histogram equalization, the cumulative histogram of the image is derived from the statistical histogram, whereas in platform histogram equalization, the cumulative histogram of the image is derived from the platform histogram. Subsequently, the gray levels of the image are redistributed through the cumulative histogram to produce an equalized image:

$$F_T(k) = \sum_{j=0}^k P_T(j), (0 \leq k \leq 255) \quad (12)$$

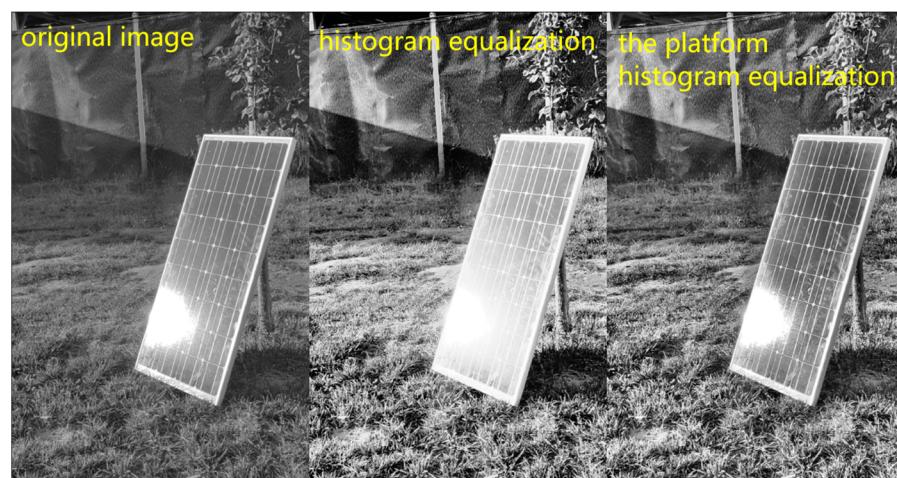
$$D_T(k) = \frac{255F_T(k)}{F_T(255)} \quad (13)$$

where  $F_T(k)$  represents the cumulative histogram of the image and  $D_T(k)$  represents the grayscale value of pixels with a grayscale level of  $k$  after platform histogram equalization ( $0 \leq D_T(k) \leq 255$ ). The grayscale histogram of the optimized platform histogram equalization algorithm is calculated to find the peak value, and the highest peak value is selected as the initial cluster center. The formula for calculation is as follows:

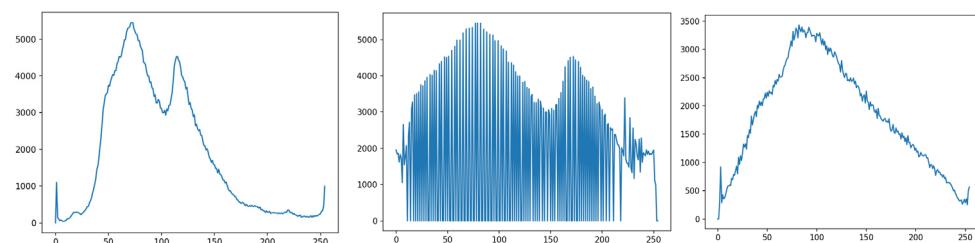
$$P_s = ((i, h_d(i)) | h_d(i) > h_d(i-1) \& h_d(i) > h_d(i+1)) \quad (14)$$

$$V_s = ((i, h_d(i)) | h_d(i) < h_d(i-1) \& h_d(i) < h_d(i+1)) \quad (15)$$

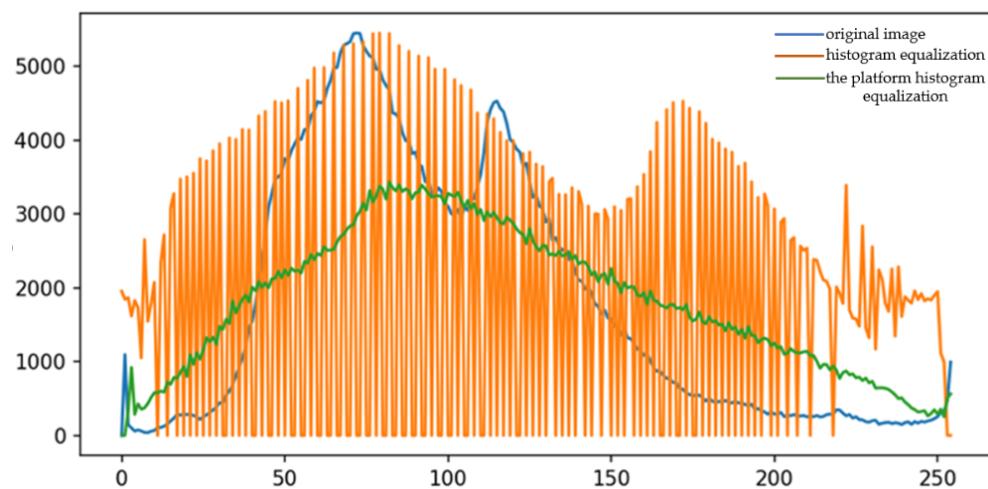
In the above equation,  $h_d(i)$  represents the histogram, and  $i$  is the grayscale value. The following images illustrate the effect of the optimization process. Figure 1 shows the original image, the image after histogram equalization, and the image after optimization using the platform histogram equalization proposed in this paper from left to right, and Figure 2 represents the corresponding grayscale histograms. Figure 3 presents a comprehensive comparison of different histograms. The images processed with the platform histogram equalization proposed in this paper have improved contrast, especially in the darker regions.



**Figure 1.** Original image and the images after different histogram equalization.

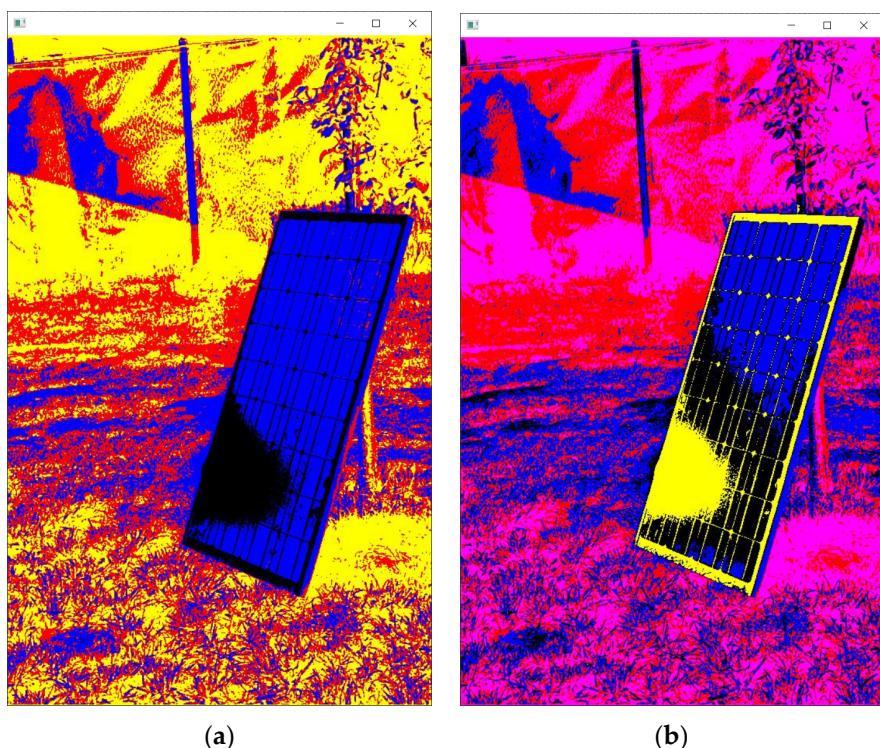


**Figure 2.** The corresponding grayscale histograms.



**Figure 3.** A comprehensive comparison of different histograms.

By using the optimized histogram processing method to select the peak grayscale as the clustering center, the Meanshift clustering method mentioned earlier is optimized. Figure 4 compares the original image to the effect of Meanshift clustering with the optimized initial cluster center selection method proposed in this paper, demonstrating that the optimization of the initial cluster centers results in improved clustering performance.



**Figure 4.** Comparison of the original and improved clustering methods for image segmentation: (a) mean-shift clustering of the original image and (b) the improved mean-shift clustering method.

## 2.2. Adaptive Shi-Tomasi Algorithm

### 2.2.1. Core Principles of the Shi-Tomasi Algorithm

The Shi-Tomasi algorithm is an enhancement of the classical corner detection algorithm, the Harris algorithm. Generally, it yields superior corners compared with the Harris algorithm. In this section, we will briefly delineate the theoretical underpinnings of the Shi-Tomasi algorithm. The core of the Harris algorithm involves utilizing a local window to

move across the image, assessing whether there has been a significant change in grayscale values. If the grayscale values within the window (as represented on the gradient map) exhibit substantial variations, then there exists a corner in the region where this window is located.

Initially, by establishing a mathematical model, it is determined which windows would cause significant changes in grayscale values. By positioning the center of a window at a location in the grayscale image, the pixel grayscale value at that position serves as the starting value. If this window is shifted by a small displacement in both the x- and y-directions to a new position, the pixel grayscale value at that position represents the change in grayscale values caused by the window movement. Assuming the simplest case, where all the pixels within the window are assigned a weight of 1, representing an average filtering kernel, the formula for the change in pixel grayscale values caused by moving the window in various directions is as follows:

$$E(u, v) = \sum_{x,y} w(x, y)[\Gamma(x + u, y + v) - I(x, y)]^2 \quad (16)$$

After expanding using Taylor's formula, the approximation is given by:

$$E \approx [u, v] \sum w(u, v) \begin{bmatrix} I_x^2, I_x I_y \\ I_x I_y, I_y^2 \end{bmatrix} \begin{pmatrix} u \\ v \end{pmatrix} \quad (17)$$

For small local displacements  $[u, v]$ , the following expression can be approximately obtained:

$$E \approx [u, v] M \begin{pmatrix} u \\ v \end{pmatrix} \quad (18)$$

$M$  is a  $2 \times 2$  matrix obtained from the derivatives of the image:

$$M = \sum_{x,y} w(u, v) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (19)$$

After diagonalizing the matrix  $M$ , the eigenvalues  $\lambda_1$  and  $\lambda_2$  represent the grayscale change rates in the X- and Y-directions, respectively.

$$M = \sum_{x,y} w(u, v) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (20)$$

The corner response function for the Harris corner detection algorithm is:

$$R = \lambda_1 \lambda_2 - K(\lambda_1 + \lambda_2)^2 \quad (21)$$

The Harris corner detection algorithm involves thresholding the corner response function  $R$ :  $R > \text{threshold}$ , which identifies local maxima in  $R$ . Shi-Tomasi's algorithm is an improvement on Harris, where, similar to Harris, if the value of the minimum eigenvalue ( $\lambda_1$  or  $\lambda_2$ ) exceeds a minimum value, the point is considered a corner [19].

The corner response function of the Shi-Tomasi corner detection algorithm is:

$$R = \min(\lambda_1, \lambda_2) \quad (22)$$

### 2.2.2. Adaptive Threshold Shi-Tomasi Algorithm

Based on the previous section, both the Harris and Shi-Tomasi corner detection algorithms require the manual setting of a final corner feature value filtering threshold. This subsection, building upon the Shi-Tomasi corner detection algorithm, proposes a multi-level thresholding technique to achieve adaptive threshold setting. Given an image, let  $I(i, j)$  denote the gray value of the pixel at position  $(i, j)$ , where  $1 \leq i \leq M$  and  $1 \leq j \leq N$ , and  $0 \leq l(i, j) \leq L - 1$ , where  $L$  is the grayscale level of the image. First,

the grayscale histogram of the image is computed. It is assumed that an optimal threshold  $T$  can be found to segment the image into a binary image. In the binary image, all pixels with grayscale values lower than  $T$  are replaced with  $A$ , and all pixels with grayscale values higher than  $T$  are replaced with  $B$ . Defining the error function as the sum of the squared differences between the gray values of corresponding pixels in the binary image and the original image, and employing integration for ease of expression, the error function can be represented as:

$$e^2 = \int_0^T (k - A)^2 h(k) dk + \int_T^{L-1} (k - B)^2 h(k) dk \quad (23)$$

In Equation (23),  $L$  represents the gray value of the pixel, and  $h(k)$  denotes the frequency of the gray value in the histogram. By setting the partial derivatives of  $e$  with respect to  $T$ ,  $A$ , and  $B$  equal to 0, one can derive:

$$T = \frac{A + B}{2} \quad (24)$$

$$A = \frac{\int_0^T k \cdot h(k) dk}{\int_0^T h(k) dk} = \mu_1 \quad (25)$$

$$B = \frac{\int_{T+1}^{L-1} k \cdot h(k) dk}{\int_{T+1}^{L-1} h(k) dk} = \mu_2 \quad (26)$$

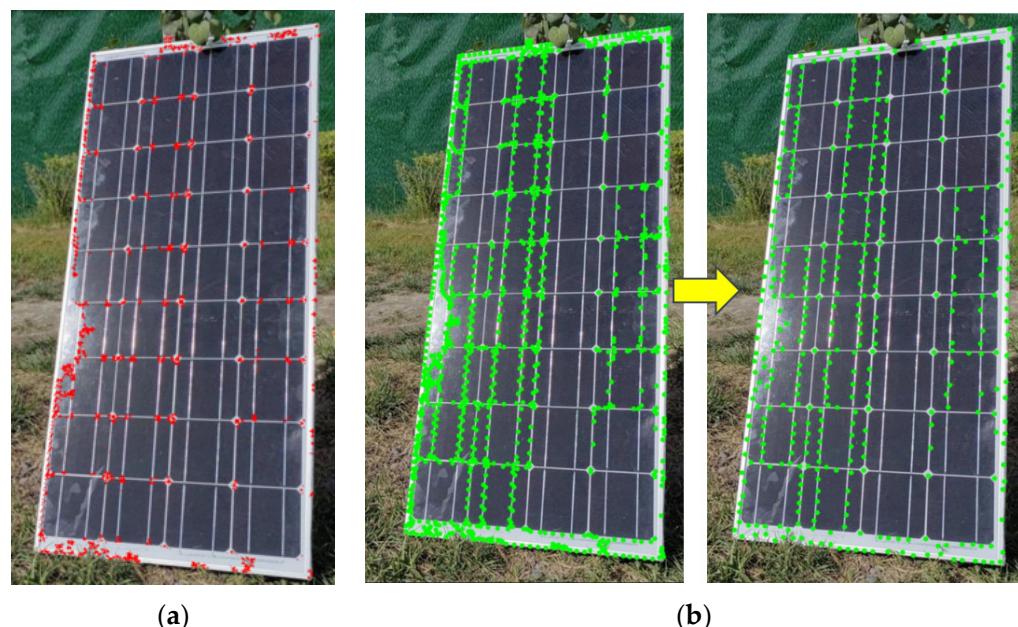
$A$  and  $B$  represent the means of the two parts of the histogram divided by the threshold  $T$ . It is noteworthy that the threshold  $T$  is determined solely by the means  $A$  and  $B$  of these two parts. However, the calculation of these means  $A$  and  $B$  is only possible once the threshold  $T$  has been established. Therefore, an iterative algorithm is required as follows: an initial threshold is first selected as a starting point; then, using this threshold, the histogram is divided into two parts, and the means of each part are calculated separately. Subsequently, the threshold is updated to be half of the sum of the means of these two parts. This process is repeated until the threshold converges. The steps involved are as follows:

- (1) Choose an initial threshold (the mean of the entire histogram).
- (2) Divide the histogram into two parts using this threshold, compute the means of the two parts, and take the average as the updated threshold.
- (3) Repeat the above process until the threshold converges.

Based on this concept, the Meanshift algorithm is initially employed to determine the underlying probability density mode number,  $K$ . The threshold between adjacent modes is calculated through an iterative threshold selection method. Subsequently, a multi-level thresholding approach (with  $K$  thresholds) is utilized to segment the image grayscale range into  $K + 1$  parts. These  $K$  thresholds effectively segment the image histogram into  $K + 1$  non-overlapping regions according to the following formula:

$$J(i, j) = \begin{cases} 0 & , T(1) \leq I(i, j) \leq T(2) \\ \frac{L-1}{k+1}, & T(2) \leq I(i, j) \leq T(3) \\ \frac{2(L-1)}{k+1}, & T(3) \leq I(i, j) \leq T(4) \\ \dots \\ L-1, & T(K+1) \leq I(i, j) \leq T(K+2) \end{cases} \quad (27)$$

The final step is to set the iteratively determined thresholds as the Shi-Tomasi algorithm thresholds for the given image, enabling adaptive thresholding. The specific effect is shown in Figure 5 below.



**Figure 5.** Comparison of the Harris algorithm and the adaptive threshold Shi-Tomasi algorithm: (a) the Harris algorithm and (b) the adaptive threshold Shi-Tomasi algorithm.

As can be seen from the above Figure 5, the Shi-Tomasi algorithm has better edge detection accuracy compared with the Harris algorithm, and the adaptive threshold Shi-Tomasi algorithm has better quality and quantity of detection points compared with the original algorithm.

### 2.3. Image Processing Strategy in This Paper

This study employs the platform histogram equalization algorithm described in Section 2.1.3 to enhance the captured images of photovoltaic panels. Furthermore, the initial clustering center of the clustering algorithm is determined based on the peak value calculation formula, thereby enhancing the original mean-shift algorithm. Subsequently, the enhanced mean-shift algorithm is employed for the preliminary segmentation of complex backgrounds and target photovoltaic panels, removing most of the background areas unrelated to the target object. Then, the adaptive Shi-Tomasi algorithm described in Section 2.2 is utilized to optimize feature point extraction for the preliminarily segmented target area, generating contour points from the edge feature points. This approach achieves precise segmentation and contour extraction for photovoltaic panels under various environments. The comprehensive image segmentation strategy is outlined as follows:

Step 1: Use the platform histogram to determine the initial clustering centers for the Meanshift algorithm.

Step 2: Perform clustering operations using the Meanshift algorithm as per Equation (6).

Step 3: Apply multi-level thresholding based on Equation (27) to the target area obtained after clustering.

Step 4: Utilize the Shi-Tomasi algorithm with the threshold values obtained through the iterative algorithm to extract feature points.

Step 5: Finally, generate image contours based on the obtained feature points to complete the image segmentation.

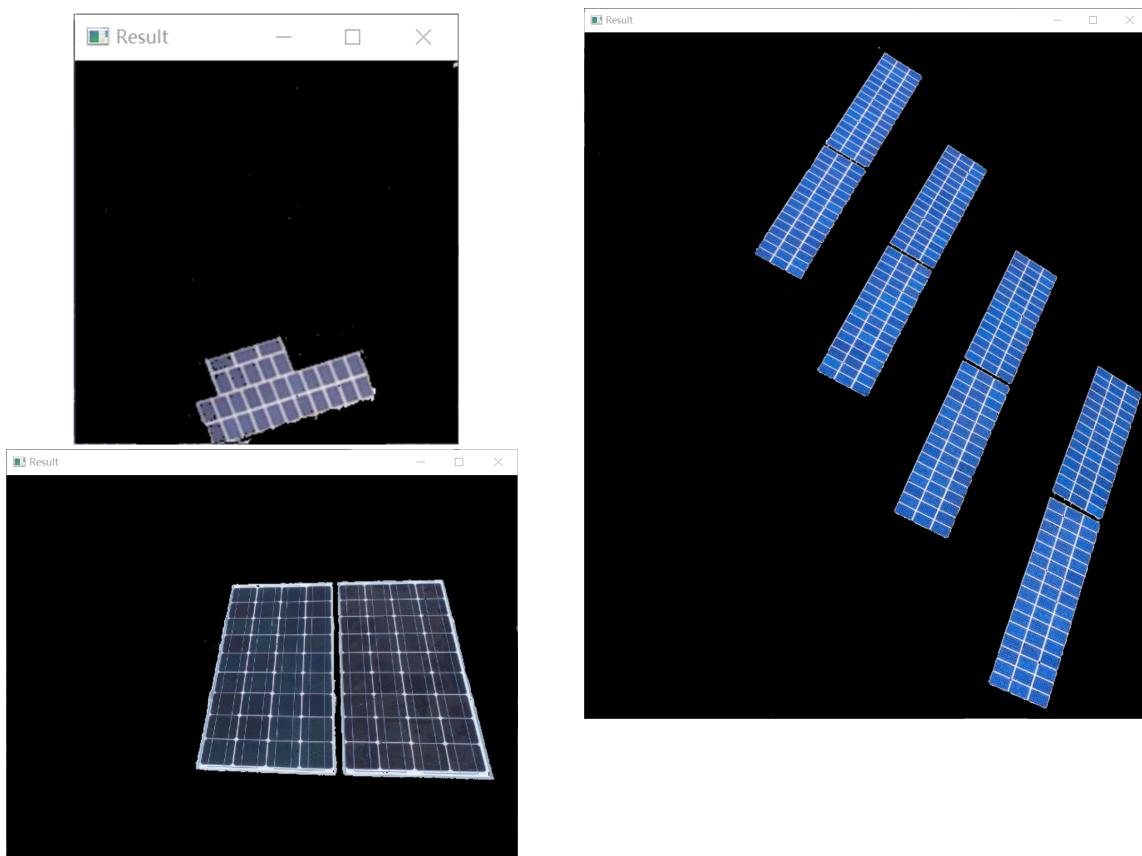
### 3. Results

To validate the segmentation efficacy of the algorithm presented in this paper, experiments were conducted using OpenCV and Python tools on photovoltaic panel images under various environmental conditions. The experimental data were acquired from a camera platform with the model number MER-125-30 UC. The algorithm was executed on a CPU system with a clock frequency of 3.3 GHz and 4 GB of RAM, operating under the Windows 10 environment. The iteration error of the algorithm was set to  $1 \times 10^{-5}$ . Furthermore, as per reference [19], the Gaussian kernel function's coefficient was set to  $\sigma = 6.5$ , the neighborhood window size was  $3 \times 3$ , and the peak point after histogram smoothing was determined to be the image's center point. The semantic segmentation model of the convolutional neural network (CNN) in this experiment was the Fully Convolutional Network (FCN), a classic model in the field of deep neural network semantic segmentation. The FCN model for this experiment is based on the VGG-16 model, specifically structured as follows: convolutional layer 1 (conv3-64), convolutional layer 2 (conv3-128), convolutional layer 3 (conv3-256), and convolutional layer 4 (conv3-512), each with 64, 128, 256, and 512  $3 \times 3$  convolutional kernels, respectively. In between each two layers, there is a maxpooling layer with a  $2 \times 2$  kernel and a stride of 2. After convolutional layer 5 (conv3-512), there are 6, 7, and 8 convolutional layers, with the corresponding kernel sizes being (7, 7, 4096), (1, 1, 4096), and (1, 1, 1000) respectively. Since all layers in the network are convolutional, it is also referred to as a fully convolutional neural network, with a final soft-max prediction layer. The semantic segmentation algorithm based on convolutional networks utilized 200 images from the dataset as the test set, while the remaining images served as the training set and validation set. The experiments primarily focused on the following three typical scenarios involving photovoltaic panel deployments: a grassland background, a sandy soil background, and a brick roof background. The dataset for these experiments comprised 2400 images from each scenario, totaling 800 images per dataset. In this paper, a representative image from each dataset was selected for visual qualitative analysis, and quantitative calculations were performed on the entire dataset of 2400 images. The accuracy of different algorithms was evaluated using the F1-score metric [20] across various environmental datasets, while the boundary accuracy was assessed using the Hausdorff distance metric [21].

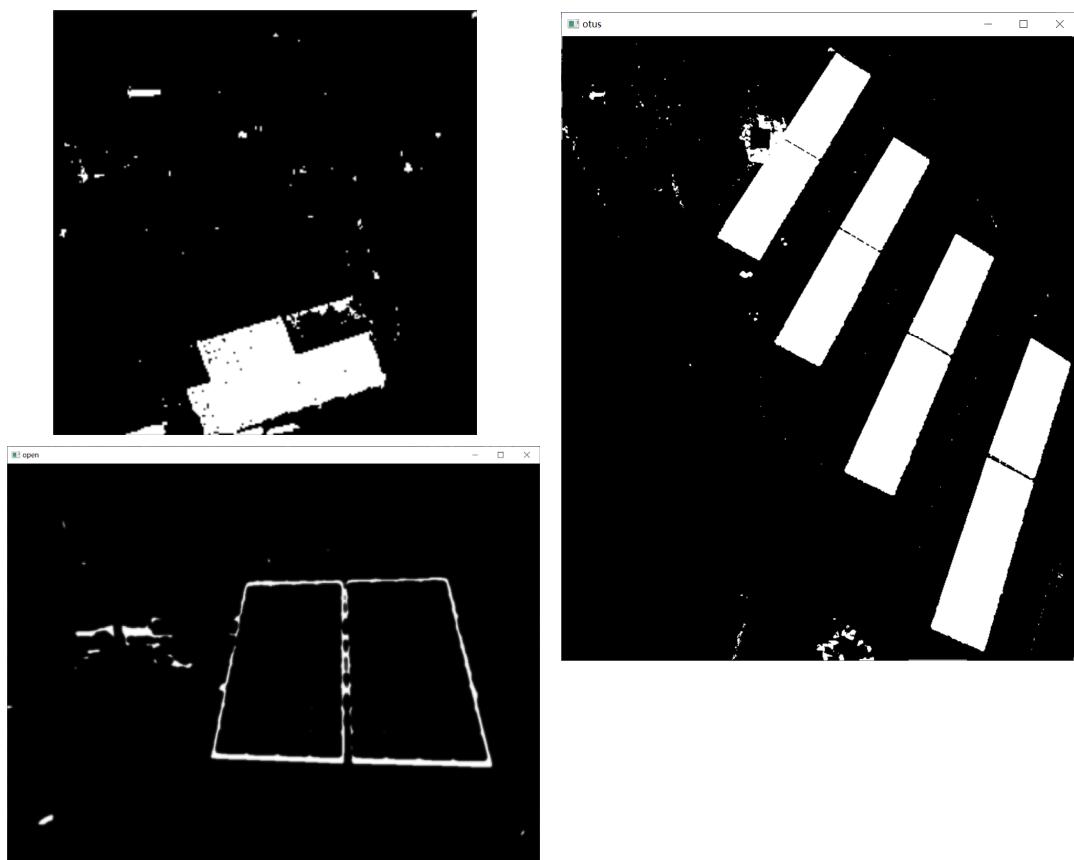
The present study employs the Otsu algorithm, the K-means algorithm, the algorithm from reference [12], the semantic segmentation algorithm of convolutional neural networks, and the segmentation strategy introduced in this paper for testing and comparison. To validate the performance of the algorithms, experiments were conducted on photovoltaic panels set against backgrounds including grasslands, brick roofs, and sandy soil. A large number of images were employed in the experimental process. In Figures 6–11, representative images are selected for intuitive description and display. Meanwhile, in Figures 12–18, this paper conducts quantitative calculations on all experimental images, which are displayed in the form of scatter plots.



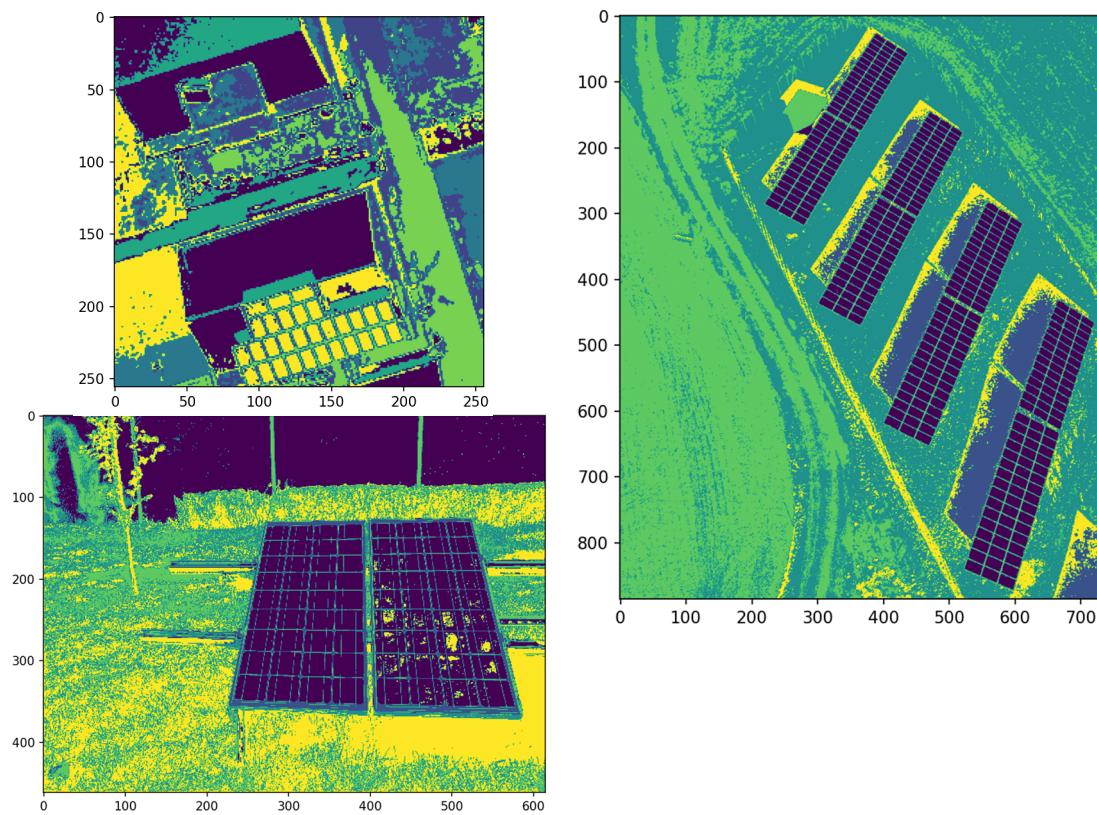
**Figure 6.** Original images.



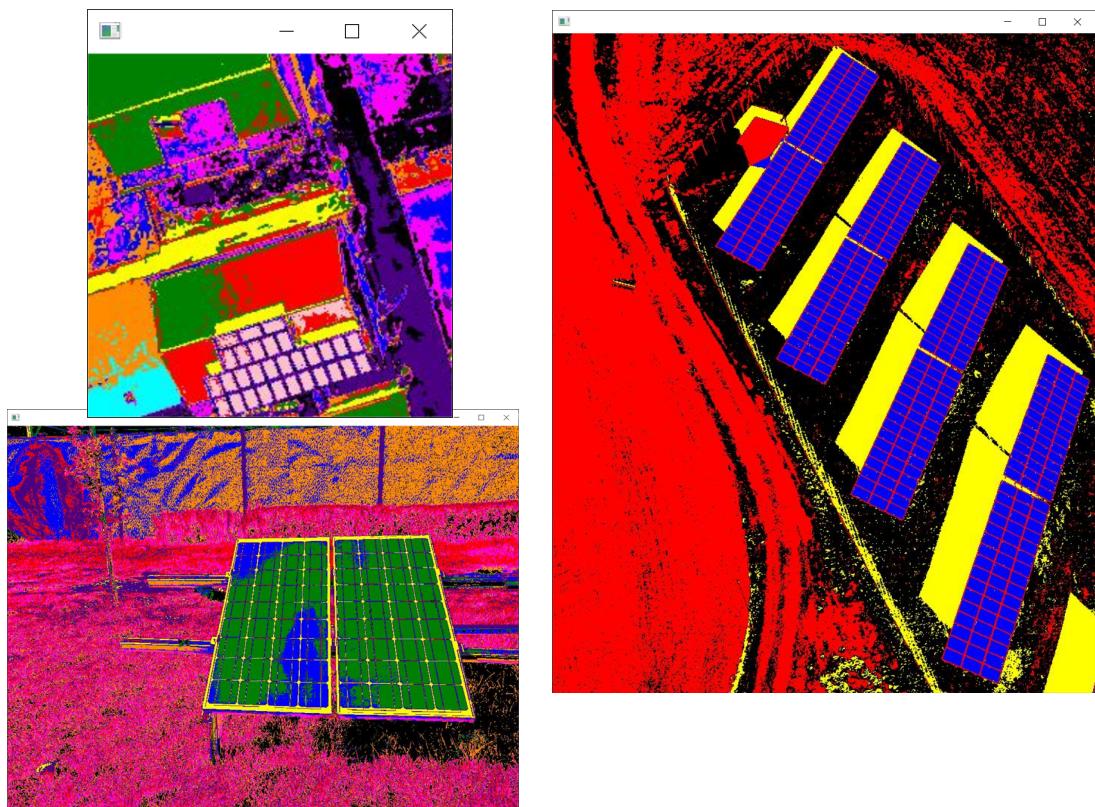
**Figure 7.** Manual segmentation of images.



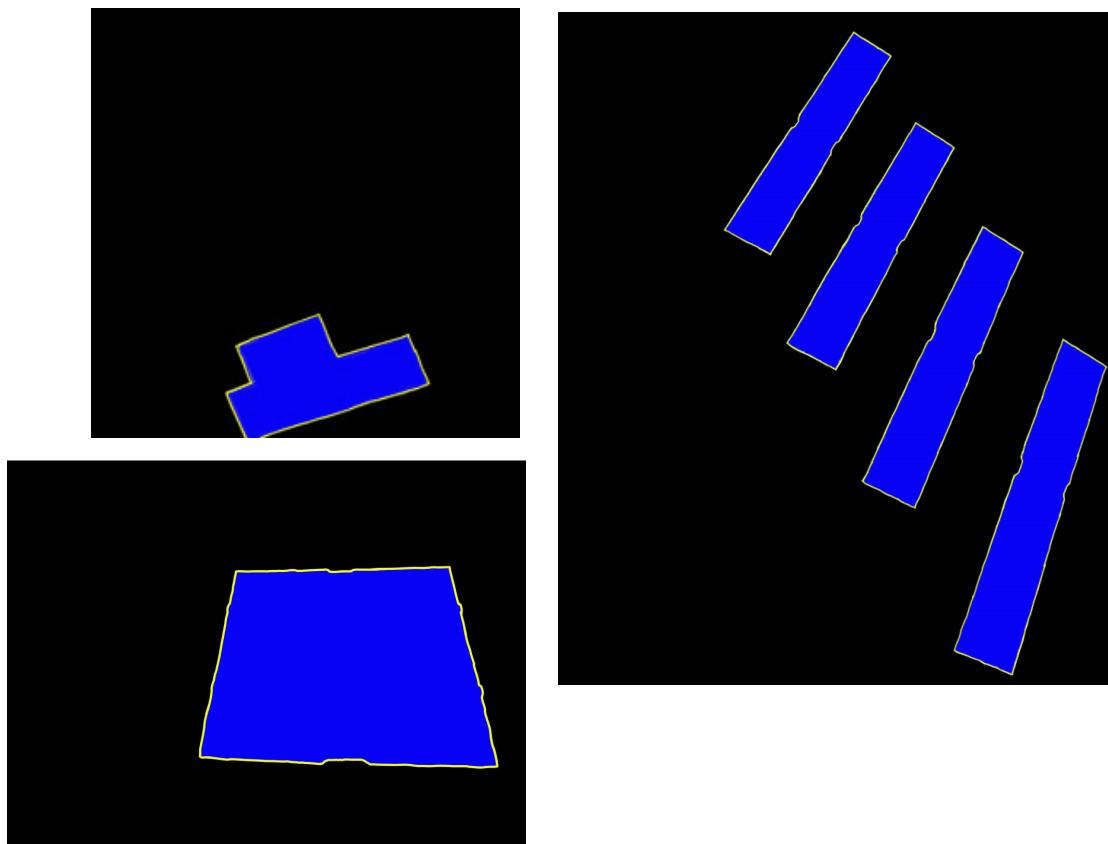
**Figure 8.** Results of the Otsu algorithm.



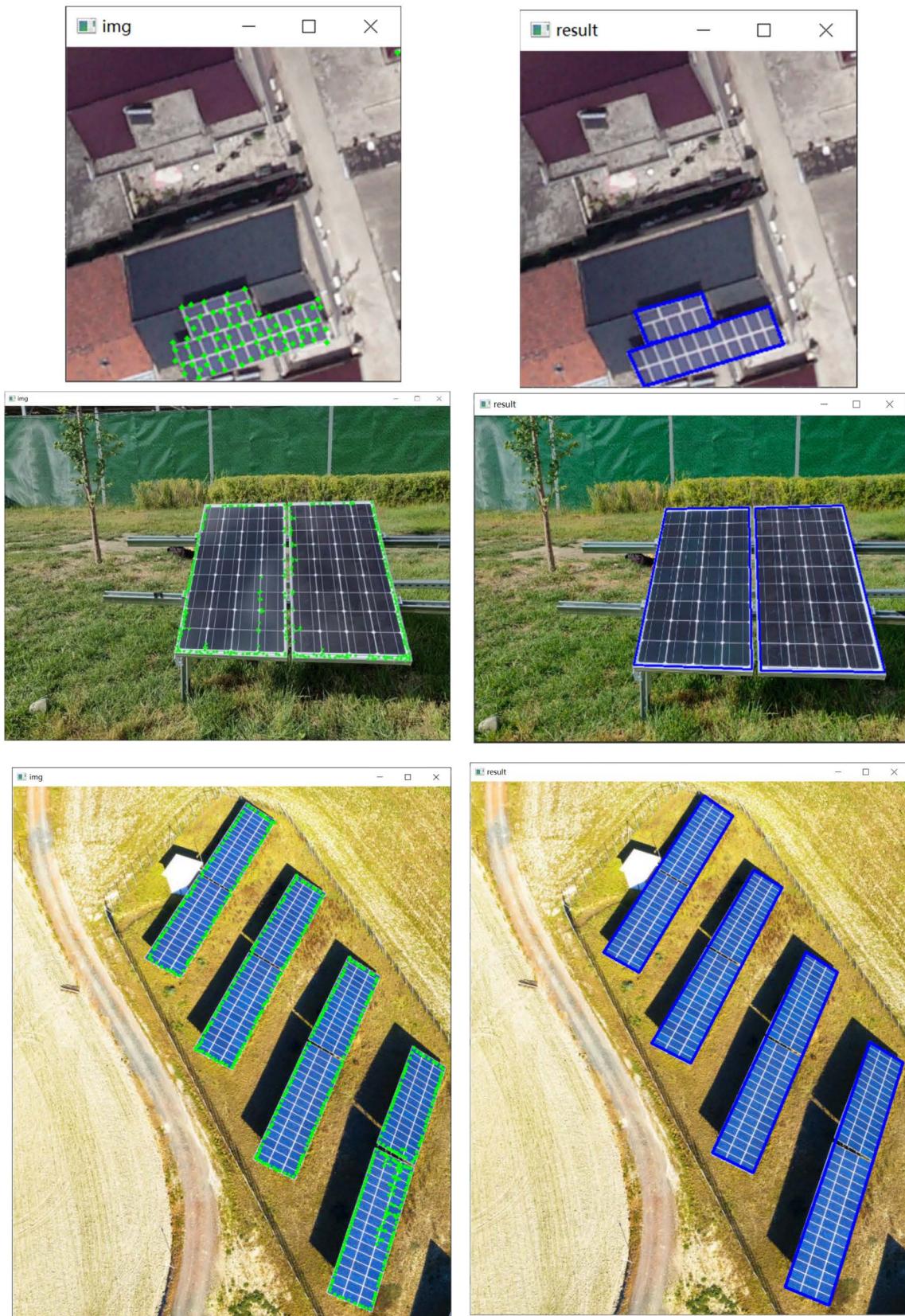
**Figure 9.** Results of the K-means algorithm.



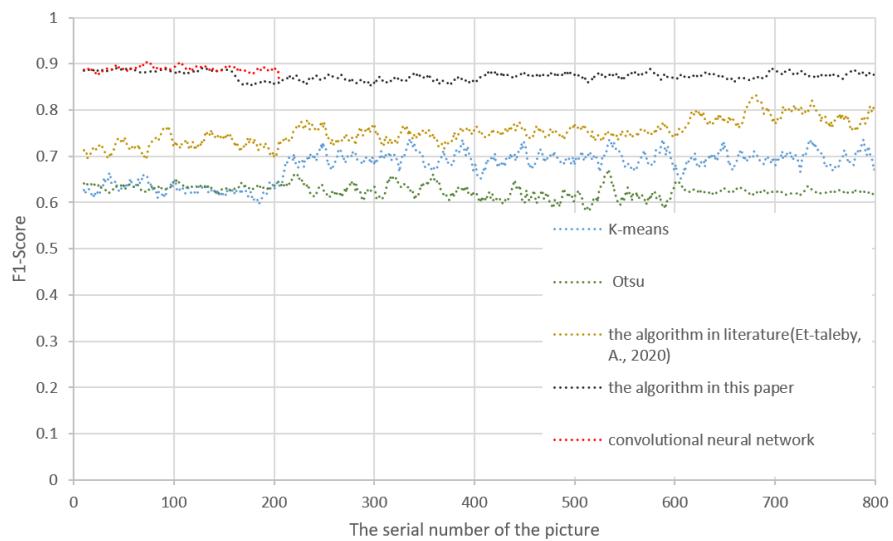
**Figure 10.** Results of the algorithm in reference [12].



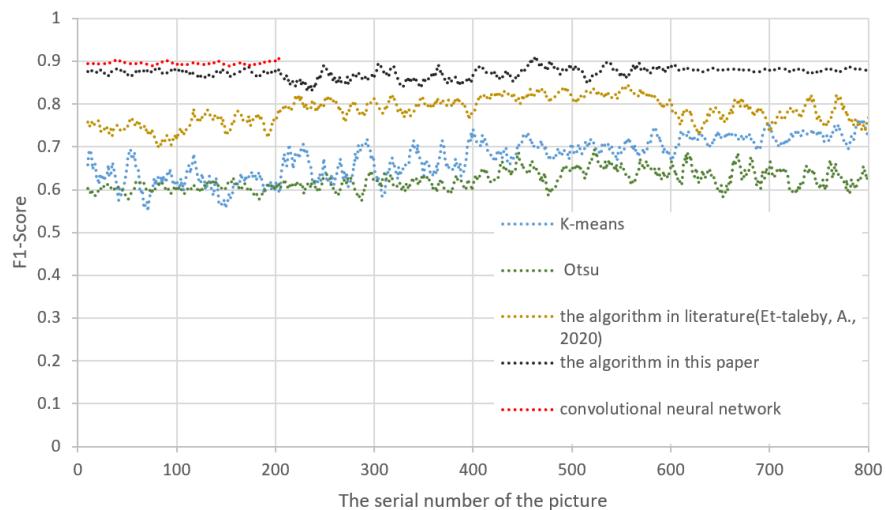
**Figure 11.** Results of the semantic segmentation algorithm.



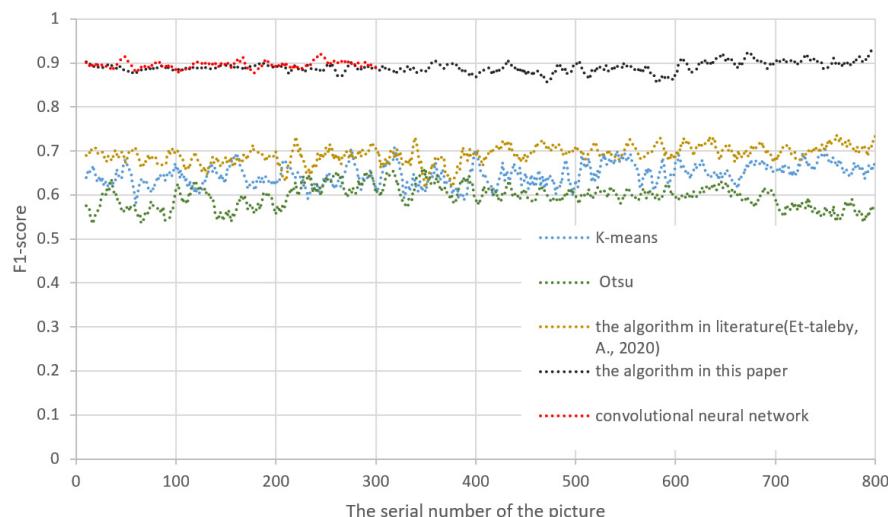
**Figure 12.** Results of the algorithm in this paper.



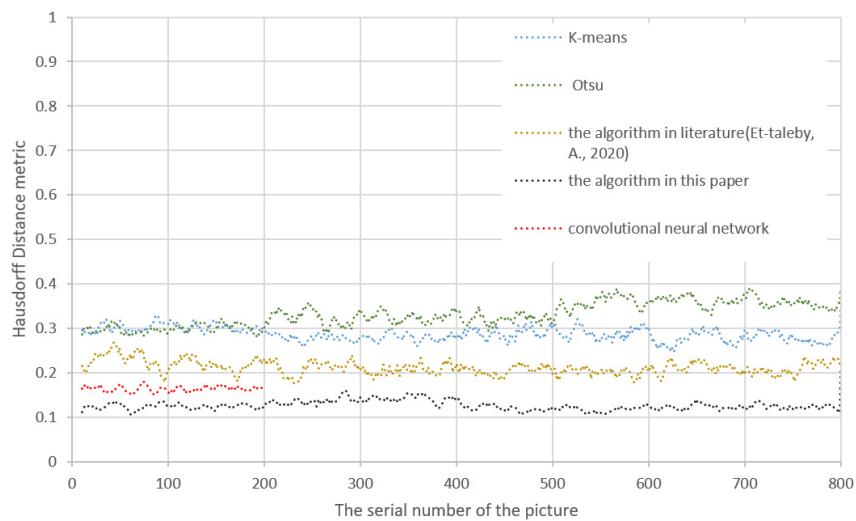
**Figure 13.** F1-score values of various algorithms against the grassland background [12].



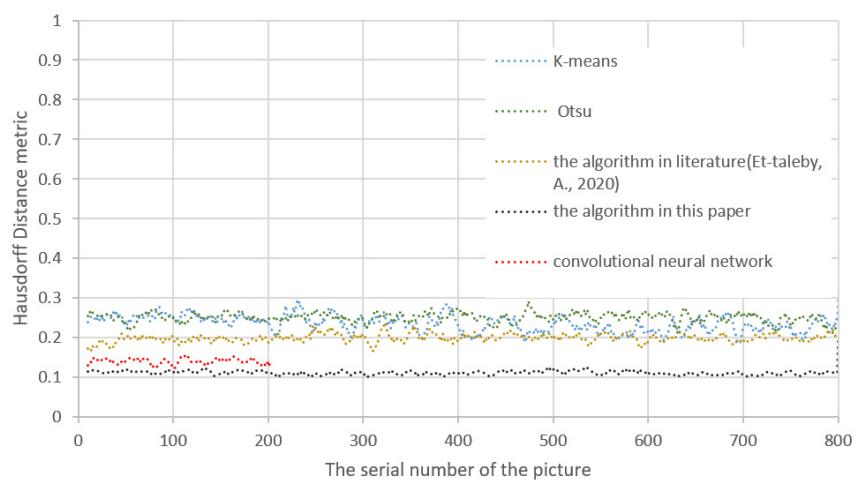
**Figure 14.** F1-score values of various algorithms against the sandy soil background [12].



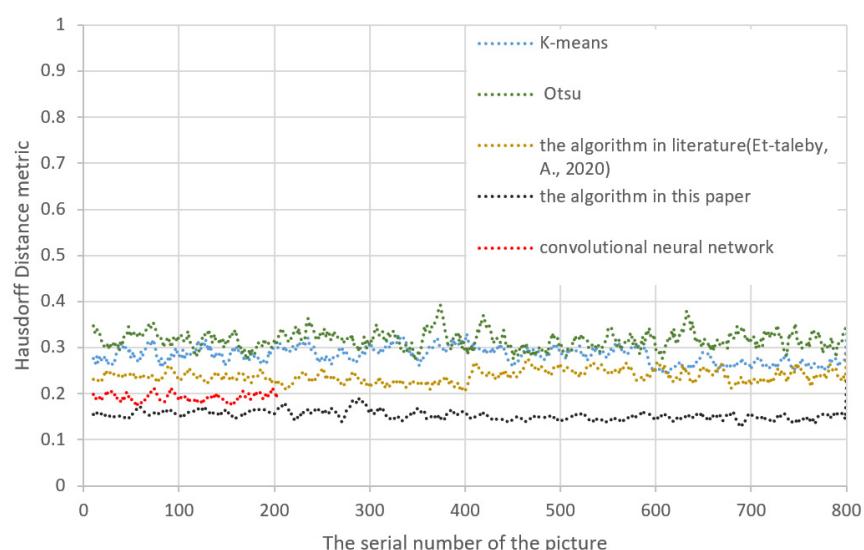
**Figure 15.** F1-score values of various algorithms against the brick roof background [12].



**Figure 16.** Hausdorff distance metric of various algorithms against the grassland background [12].



**Figure 17.** Hausdorff distance metric of various algorithms against the sandy soil background [12].



**Figure 18.** Hausdorff distance metric of various algorithms against the brick roof background [12].

#### 4. Discussion

This study selects three major typical scenarios for the application of photovoltaic panels for experimental validation. Meanwhile, two major evaluation metrics including the F1-score and Hausdorff distance are employed to compare and assess the algorithms. Before interpreting these evaluation metrics, it is essential to clarify the concept of the confusion matrix, as these metrics are related to or derived from the confusion matrix. The confusion matrix, also known as the error matrix, is a standard format for representing precision metrics, represented in a matrix format with n rows and n columns. Specific evaluation metrics include overall precision, mapping precision, user precision, etc. These precision metrics reflect the accuracy of image classification from different perspectives. In the evaluation of image accuracy, it is mainly used to compare the classification results with the actual measured values. The accuracy of the classification results can be displayed in a confusion matrix, as shown in Figure 19:

Confusion Matrix		
<i>Actual</i>	0	1
<i>Predict</i>		
0	TN	FN
1	FP	TP

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1 Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

**Figure 19.** Confusion matrix.

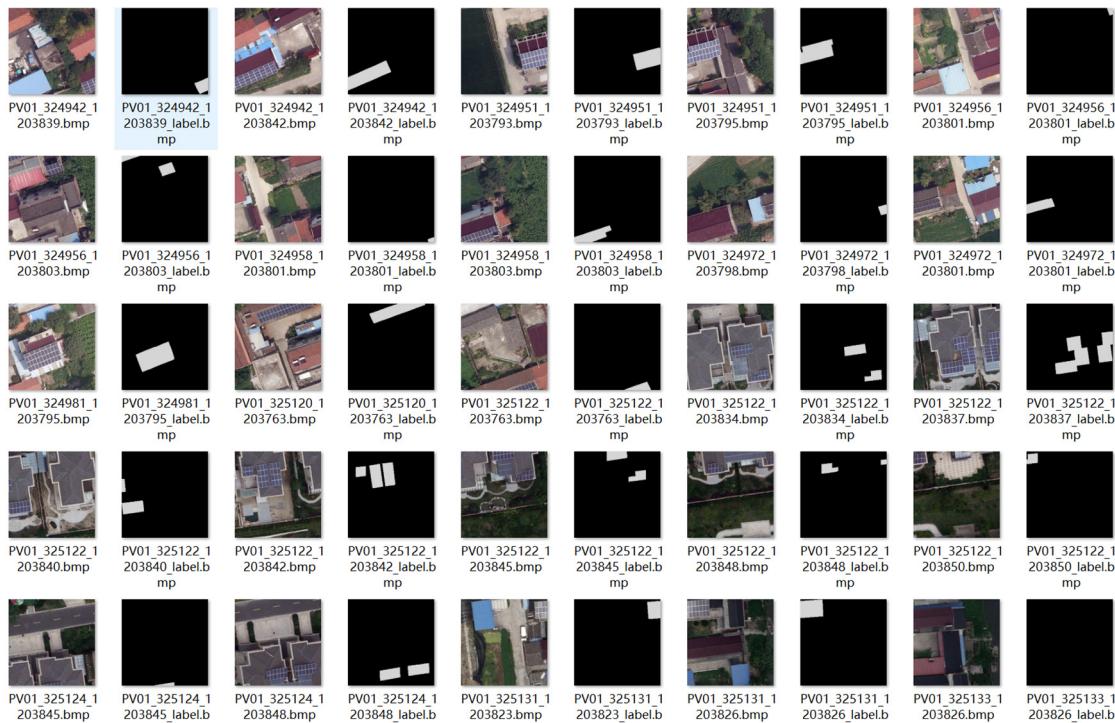
In the figure, TP (True Positive) is determined as a positive sample, which is a positive sample; TN (True Negative) is a sample judged as negative that indeed is negative; FP (False Positive) is an instance that is classified as a positive sample but is a negative sample; and FN (False Negative) is a sample judged as negative but in fact is positive.

The F1-score is an index used in statistics to evaluate the precision of binary classification models. It simultaneously considers both the precision and recall rates of the classification model. The F1-score can be viewed as a weighted average of the model's precision and recall rates, with a maximum value of 1 and a minimum value of 0, as shown in the formula in Figure 19. In this study, all images within the dataset were manually segmented into precise regions, serving as the positive reference set for the experimental data, as shown in Figure 20. Additionally, their contour coordinate sets were acquired as the boundary-positive reference set.

This study employs the Otsu algorithm, the K-means algorithm, the algorithm from reference [12], and a fusion of the Meanshift and Shi-Tomasi algorithms for testing and comparison to validate the performance of the algorithms. The validation was conducted on photovoltaic panels set against backgrounds of grasslands, brick roofs, and sandy soil. A large number of images were employed in the experimental process, and several representative images were selected for illustration and demonstration in this paper.

The visual effects of various algorithms for segmentation under different backgrounds can be seen in Figures 7–12. Figure 6 shows the original image of a single photovoltaic panel image in a complex environment, Figure 7 is the manual segmentation of the photovoltaic panel image, Figure 8 is the segmentation result of the Otsu algorithm, Figure 9 is the segmentation result of the K-means clustering algorithm, Figure 10 is the segmentation result of the algorithm in reference [12], and Figure 11 is the segmentation result of the algorithm presented in this paper. From the figures above, it can be observed that the Otsu algorithm in Figure 8 cannot effectively differentiate objects similar to the edges of photovoltaic panels in a complex background compared with manual segmentation, and the contour generation is incomplete. The K-means clustering algorithm in Figure 11

cannot stably differentiate objects similar to the color and shape of photovoltaic panels in a complex background compared with manual segmentation. The algorithm in reference [12] is an improvement to the K-means clustering algorithm, and compared with the segmentation result in Figure 9, it can better differentiate areas with background colors similar to photovoltaic panels, but there are still some cases of false segmentation. Figure 11 shows the semantic segmentation results of a neural network, and Figure 12 shows the segmentation process and results of the algorithm presented in this paper. Compared with the results of the above algorithms, the semantic segmentation results of the neural network more closely match the manual segmentation areas compared to the segmentation contour areas generated by the algorithm presented in this paper.



**Figure 20.** Positive sample reference dataset.

Next, we employ the F1-score metric to quantitatively analyze the regional segmentation accuracy of algorithms across various backgrounds. Based on the F1-score values of 800 images under the grassland background shown in Figure 13, it is evident that the integrated segmentation strategy proposed in this paper, which combines the Meanshift and Shi-Tomasi algorithms, outperforms traditional Otsu threshold segmentation, K-means clustering, and the algorithm described in reference [12]. By averaging all F1-score values for each algorithm in the grassland background, Table 1 is obtained. Notably, the segmentation method proposed in this paper exhibits an average improvement of 30% over traditional Otsu threshold segmentation, K-means clustering, and the algorithm described in reference [12], and it is only 0.45% inferior to the accuracy of convolutional neural network semantic segmentation. In a sandy soil background, as depicted in Figure 14, the proposed segmentation method continues to offer significant advantages over traditional Otsu threshold segmentation, K-means clustering, and the algorithm described in reference [12], achieving an average improvement of 26.2%. However, it is only 2.87% inferior to the accuracy of convolutional neural network semantic segmentation. In the most complex setting of a brick roof background, the accuracy rates of all algorithms exhibit some decline. However, referring to Table 1, the algorithm proposed in this paper still offers an average improvement of 39.4% over traditional Otsu threshold segmentation, K-means clustering, and the algorithm described in reference [12], while achieving a precision comparable to that of the convolutional neural network semantic segmentation algorithm.

**Table 1.** Comparison of algorithm results.

Background	Parameters	Method				
		Otsu	K-Means	Reference [12]	Semantic Segmentation	Algorithm Presented in This Paper
grassland	F1-score	0.642	0.684	0.739	0.891	0.887
	Hausdorff-95	0.342	0.288	0.246	0.163	0.121
sandy soil	F1-score	0.625	0.687	0.761	0.897	0.872
	Hausdorff-95	0.268	0.251	0.193	0.137	0.113
brick roof	F1-score	0.574	0.642	0.708	0.897	0.894
	Hausdorff-95	0.326	0.285	0.241	0.198	0.165

After analyzing the accuracy of target region segmentation using the F1-score, the accuracy of boundary contour segmentation is also crucial. In this paper, the Hausdorff distance is employed to evaluate the algorithm's accuracy in boundary contour segmentation. Unlike the aforementioned Dice coefficient (of which the F1-score is a type), the Dice coefficient is sensitive to the internal filling of the image, while the Hausdorff distance is sensitive to the boundaries of the segmented region.

This study employs the Hausdorff distance method to evaluate the accuracy of boundary contour segmentation for various algorithms under three typical backgrounds. A closer value to zero indicates a higher similarity between the contour and the reference positive sample set, indicating superior segmentation accuracy. Based on the Hausdorff distance values for different algorithms under the grassland, sandy soil, and brick roof backgrounds depicted in Figures 16–18, it is evident that the proposed comprehensive segmentation strategy exhibits superior accuracy compared with traditional Otsu threshold segmentation, K-means clustering, and the algorithm mentioned in reference [12]. The average Hausdorff distance of all images under different algorithms was also calculated. Referring to Table 1, in the grassland background, the segmentation method proposed in this paper, compared with the traditional Otsu threshold segmentation algorithm, K-means clustering algorithm, and the algorithm described in reference [12], exhibits an average improvement of over 58.5% in boundary contour segmentation accuracy and is superior to the convolutional neural network semantic segmentation algorithm by 25.8%. In the sandy soil background, the proposed segmentation method still has a significant advantage over the traditional Otsu threshold segmentation algorithm, K-means clustering algorithm, and the algorithm described in reference [12], with an average improvement of 52.4%, and outperforms the convolutional neural network semantic segmentation algorithm by 17.5%. In the most complex background of brick roofs, the algorithm proposed in this paper still shows an average improvement of 41.9% compared with the traditional Otsu threshold segmentation algorithm, K-means clustering algorithm, and the algorithm described in reference [12] and is superior to the convolutional neural network semantic segmentation algorithm by 20% in accuracy.

The execution time or speed of an algorithm is a crucial metric, particularly during the inference phase after the algorithm model is deployed. This evaluation uses average execution time to assess each algorithm by calculating the total time executed across a full sample set to determine the average execution time. Furthermore, the average execution time of the algorithm in this paper is used as the baseline for comparison, as shown in the Table 2 below.

From the Table 2, it is evident that the algorithm proposed in this study exhibits a 37.67% advantage in execution time over the neural network semantic segmentation algorithm under the conditions of this experiment. Furthermore, it outperforms the method described in reference [12] by 18.8%. In terms of execution time, the algorithm is comparable to the K-means algorithm. After analyzing the algorithm performance in three typical scenarios, it is evident that the algorithm proposed in this paper possesses significant advantages over traditional threshold and clustering segmentation algorithms in terms of segmentation accuracy within the target region. It is only slightly inferior to convolutional

neural network semantic segmentation algorithms. However, when it comes to accurate segmentation of the target contour, the proposed algorithm shows pronounced advantages in segmentation efficiency. Based on the aforementioned experimental comparisons, the algorithm presented in this paper exhibits commendable segmentation performance across three typical application scenarios for photovoltaic panels. Furthermore, within the application scenarios discussed in this study, the algorithm can also be extended to objects with similar shapes, enabling broader applications.

**Table 2.** Comparison of average execution time.

Method	Average Execution Time (ms)	
Baseline		84.22
Algorithm presented in this paper	94.22	+0
Otsu	87.87	-6.35
K-means	95.96	+1.74
Reference [12]	113.73	+19.51
Semantic segmentation	151.14	+56.92

## 5. Conclusions

This paper proposes a segmentation strategy that integrates the Meanshift and adaptive Shi-Tomasi algorithms. The experimental results of this paper show that compared with traditional image segmentation algorithms, the algorithm proposed in this paper achieves excellent accuracy in target region segmentation and boundary contour segmentation in typical application scenarios of photovoltaic panels. Meanwhile, in the case of small samples, compared to semantic segmentation algorithms based on convolutional neural networks, it can achieve similar accuracy in region segmentation and better accuracy in boundary contour segmentation. This method utilizes the optimized platform histogram to determine the initial center of the cluster and conducts preliminary image segmentation. Meanwhile, based on the traditional Shi-Tomasi algorithm, it adaptively obtains the threshold using a multi-level thresholding method, achieving better feature points in the target region to generate the corresponding contour. According to the experimental results, the segmentation performance of the proposed algorithm is superior to the Otsu algorithm, K-means algorithm, and the algorithm in reference [12]. In three typical application scenarios of grassland, sandy soil, and brick roofs, it achieves more than 30% improvement in segmentation accuracy of the target region of photovoltaic modules and more than 40% improvement in segmentation accuracy of the boundary contour. Compared with the commonly used convolutional neural network semantic segmentation model, it has similar target region segmentation accuracy and about 20% improvement in boundary contour segmentation accuracy. The results indicate that this image segmentation strategy has excellent segmentation capabilities in the three typical photovoltaic panel application scenarios of grassland, sandy soil, and brick roofs.

**Author Contributions:** Conceptualization, C.H. and X.C.; software design, C.H.; software validation, W.Z.; resources, L.G.; data curation, W.Z.; writing—original draft, C.H.; writing—review and editing, C.H. and X.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by Major Science and Technology Projects in Xinjiang Uygur Autonomous Region of China (No. 2022A01004-4).

**Data Availability Statement:** Data are contained within the article.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

- Yao, S.; Kang, Q.; Zhou, M.; Abusorrah, A.; Al-Turki, Y. Intelligent and data-driven fault detection of photovoltaic plants. *Processes* **2021**, *9*, 1711. [[CrossRef](#)]

2. Khalil, I.U.; Ul-Haq, A.; Mahmoud, Y.; Jalal, M.; Aamir, M.; Ahsan, M.U.; Mehmood, K. Comparative analysis of photovoltaic faults and performance evaluation of its detection techniques. *IEEE Access* **2020**, *8*, 26676–26700. [[CrossRef](#)]
3. Li, D. *Research and Application of Photovoltaic Array Fault Diagnosis Technology*; Guizhou University: Guizhou, China, 2021.
4. Bhargavi, K.; Jyothi, S. A survey on threshold based segmentation technique in image processing. *Int. J. Innov. Res. Dev.* **2014**, *3*, 234–239.
5. Maini, R.; Aggarwal, H. Study and comparison of various image edge detection techniques. *Int. J. Image Process. IJIP* **2009**, *3*, 1–11.
6. Hong, Y.Y.; Pula, R.A. Methods of photovoltaic fault detection and classification: A review. *Energy Rep.* **2022**, *8*, 5898–5929. [[CrossRef](#)]
7. Guo, X.; Wang, J.; Cheng, C. A Fusion of Hierarchical Clustering Algorithm and Graph-Based Segmentation Algorithm for Image Segmentation. *J. Natl. Univ. Def. Technol./Guofang Keji Daxue Xuebao* **2022**, *44*, 25–32.
8. Park, S.; Han, S.; Kim, S.; Kim, D.; Park, S.; Hong, S.; Cha, M. Improving unsupervised image clustering with robust learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 12278–12287.
9. Abubakar, A.; Jibril, M.M.; Almeida, C.F.M.; Gemignani, M.; Yahya, M.N.; Abba, S.I. A Novel Hybrid Optimization Approach for Fault Detection in Photovoltaic Arrays and Inverters Using AI and Statistical Learning Techniques: A Focus on Sustainable Environment. *Processes* **2023**, *11*, 2549. [[CrossRef](#)]
10. Mellit, A.; Kalogirou, S. Assessment of machine learning and ensemble methods for fault diagnosis of photovoltaic systems. *Renew. Energy* **2022**, *184*, 1074–1090. [[CrossRef](#)]
11. Al-Rahlawee AT, H.; Rahebi, J. Multilevel thresholding of images with improved Otsu thresholding by black widow optimization algorithm. *Multimed. Tools Appl.* **2021**, *80*, 28217–28243. [[CrossRef](#)]
12. Et-taleby, A.; Boussetta, M.; Benslimane, M. Faults detection for photovoltaic field based on k-means, elbow, and average silhouette techniques through the segmentation of a thermal image. *Int. J. Photoenergy* **2020**, *2020*, 6617597. [[CrossRef](#)]
13. Tsanakas, J.; Chrysostomou, D.; Botsaris, P.; Gasteratos, A. Fault diagnosis of photovoltaic modules through image processing and Canny edge detection on field thermographic measurements. *Int. J. Sustain. Energy* **2015**, *34*, 351–372. [[CrossRef](#)]
14. Jiang, L.; Su, J.; Shi, Y.; Lai, J. Wang Haining. Detection Method for Hot Spots in Photovoltaic Arrays Based on Infrared Thermal Image Processing. *Acta Energiae Solaris Sin.* **2020**, *41*, 180–184.
15. Mo, Y.; Wu, Y.; Yang, X.; Liu, F.; Liao, Y. Review the state-of-the-art technologies of semantic segmentation based on deep learning. *Neurocomputing* **2022**, *493*, 626–646. [[CrossRef](#)]
16. You, L.; Jiang, H.; Hu, J.; Chang, C.H.; Chen, L.; Cui, X.; Zhao, M. GPU-accelerated Faster Meanshift with euclidean distance metrics. In Proceedings of the 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), Los Alamitos, CA, USA, 27 June–1 July 2022; pp. 211–216.
17. Chen, Q.; He, L.; Diao, Y.; Zhang, K.; Zhao, G.; Chen, Y. A Novel Neighborhood Granular Meanshift Clustering Algorithm. *Mathematics* **2022**, *11*, 207. [[CrossRef](#)]
18. Wan, J.; Lin, S.; Mei, T.; Lin, Z.; Guo, T. Image Enhancement Algorithm for Electro-Wetting Displays Based on Image Segmentation and Dynamic Histogram Equalization. *Acta Photonica Sin.* **2022**, *51*, 240–250.
19. Patankar, S.S.; Kadam, S.G.; Jadhav, A.; Gore, M. Image Registration using Shi-Tomasi and SIFT. In Proceedings of the 2023 2nd International Conference for Innovation in Technology (INOCON), Bangalore, India, 3–5 March 2023; pp. 1–4.
20. Yacoubi, R.; Axman, D. Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models. In Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems, Online, 20 November 2020; pp. 79–91.
21. Huttenlocher, D.P.; Klanderman, G.A.; Rucklidge, W.J. Comparing images using the Hausdorff distance. *IEEE Trans. Pattern Anal. Mach. Intell.* **1993**, *15*, 850–863. [[CrossRef](#)]

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