electronics

Article

Research on Fast Multi-Threshold Image Segmentation Technique Using Histogram Analysis

Mingjin Xu [1,*], Shaoshan Chen [2], Xiaopeng Gao [1], Qing Ye [1], Yongsheng Ke [1], Cong Huo [1] and Xiaohong Liu [1]

[1] College of Naval Architecture and Ocean Engineering, Naval University of Engineering, Wuhan 430033, China
[2] Xichang Satellite Launch Center, Wenchang 571300, China
* Correspondence: xumingjinnudt@163.com

Abstract: This paper investigates a method for the multi-threshold segmentation of grayscale imaging using the local minimum points of a histogram curve as the segmentation threshold. By smoothing the histogram curve and judging the conditions, the expected peaks and valleys are identified, and the corresponding minimum points are used as segmentation thresholds to achieve fast multi-threshold image segmentation. Compared to the OTSU method (maximum between-class variance) for multi-threshold segmentation and the region growing method, this method has less computational complexity. In the recognition and segmentation process of solder pads with adhesion of underfill in LED Chips, the segmentation time is less than one percent of that of the OTSU method and the region growing method. The segmentation effect is better than the OTSU method and the region growing method, and it can achieve fast multi-threshold segmentation of images. Moreover, it has strong adaptability to the differences in the overall grayscale of images, meeting the requirements for high UPH (Units Per Hour) in industrial production lines.

Keywords: multi-threshold segmentation; OTSU algorithm; histogram; curve extremum method

Citation: Xu, M.; Chen, S.; Gao, X.; Ye, Q.; Ke, Y.; Huo, C.; Liu, X. Research on Fast Multi-Threshold Image Segmentation Technique Using Histogram Analysis. *Electronics* **2023**, *12*, 4446. https://doi.org/10.3390/electronics12214446

Academic Editor: Silvia Liberata Ullo

Received: 13 September 2023
Revised: 20 October 2023
Accepted: 25 October 2023
Published: 29 October 2023

1. Introduction

Image segmentation is an important image analysis technique in computer vision; it is commonly used in applications such as image recognition and 3D reconstruction. Its purpose is to extract meaningful features or required characteristics from images. Some image segmentation theories believe that an image is actually composed of information and noise, and the purpose of segmentation is to strip away the noise. However, there are many applications where it is necessary to segment a part of the foreground image in order to find the boundary of the scene of interest [1–5]. Segmented regions within an image share similar internal features or exhibit significant differences from other regions. For example, the most widely used medical image segmentation currently utilizes image segmentation algorithms to separate tumor regions from normal tissue regions, thereby achieving tumor localization and diagnosis. Currently, commonly used segmentation algorithms can be categorized based on the segmentation criteria: threshold-based image segmentation, such as the OTSU method [6–10]; clustering-based image segmentation, such as the fuzzy C-means algorithm [11–13]; and region growing-based image segmentation, such as the watershed algorithm [14]. The region growing method is a traditional image segmentation algorithm based on regions. Region growing starts from a set of initial seed points, and, through pre-defined region growing rules, domain pixels with similar properties to the seed points are continuously added to each seed point, and the final growth region is formed when the termination conditions for region growth are met [15,16]. In addition, the deep learning-based image segmentation, such as the visual big model SAM (segment anything model) proposed by Meta Company on 6 April 2023, can achieve image segmentation based on text instructions and other methods, similar to the interactive

segmentation, solving the difficulty of training set labeling in deep learning. It claims to be able to segment everything and quickly achieve multi-target segmentation [6–10].

Although researchers have proposed various segmentation theories and methods, there is currently no universal segmentation method, and there is no standard for selecting suitable methods in practical applications. Moreover, there is no segmentation algorithm that can achieve satisfactory segmentation results for all images. The performance of image segmentation methods is influenced by target size, mean difference, contrast, target variance, background variance, and noise. Generally, segmentation performance is still evaluated based on subjective visual effects. For grayscale image segmentation, segmentation quality evaluation is divided into subjective and objective evaluation criteria based on whether reference images are needed [17–22].

These algorithms have achieved good segmentation results when the processing speed is not a critical requirement. For instance, clustering segmentation and region growing algorithms have been widely applied in the fields of medical and remote sensing image processing [23,24]. However, the clustering algorithm and region growing algorithm belong to iterative algorithms, and as image pixels increase, the computational load will increase exponentially. They are not suitable for scenarios with high computational speed requirements and large images [25–30]. The OTSU method obtains an optimal threshold by minimizing the weighted sum of intra-class and inter-class variances in the image, thereby achieving the goal of automatically calculating the binarization threshold. Due to its ability to avoid subjective and cumbersome threshold selection operations, and its superior computational speed compared to the clustering algorithm and region growing algorithm, the OTSU method is widely used in fields such as image segmentation, character recognition, and facial recognition [31–33]. Histogram-based threshold segmentation has been widely used in applications that focus on operational efficiency due to its simple implementation and ability to compress data and reduce storage. If there are two distinct peaks separated from each other in the histogram information of the image, threshold segmentation should binarize the image based on the threshold points between the two peaks. However, in real images, there may be a wide overlapping range between the target and the background, and the image histogram will not have obvious separated peaks. Moreover, the small sample nature of pixel count in pixel estimation histogram information makes the histogram itself noisy. Therefore, threshold segmentation often faces many difficulties in practical applications [34–37]. The deep learning image segmentation technology represented by the SAM model has high adaptability. In theory, as long as sufficient training samples are provided, it can be applied where traditional image segmentation methods are applicable. The actual situation is that the SAM algorithm has achieved significant results in natural image segmentation but currently faces significant challenges in medical image processing. It has significant application value only in interactive medical image segmentation [38–42]. However, the current deep learning segmentation technology not only requires extremely high technical requirements but also requires a large number of samples for training and great computational power support, making them generally unsuitable for industrial fields that require fast segmentation. Therefore, when the speed of image segmentation is high and the computational cost is limited, it is necessary to design segmentation algorithms based on traditional features such as image pixels.

In industrial applications, such as on production lines, image segmentation often serves as the first step in image processing tasks like quality inspection. It helps avoid blind searching across the entire image during processing. Such applications often require extremely fast speed and high-accuracy processing to meet the highly demanding UPH (Units Per Hour) requirements in the industrial field. Meanwhile, due to factors such as lighting, equipment aging, and machine differences, the overall color and grayscale values of images collected at different times and production lines may have significant differences, making it impossible to predict the differences between foreground and background pixels in advance. Therefore, the image segmentation algorithms must have a certain degree of adaptability to this situation. In order to meet the requirements of processing speed

and adaptability in industrial production, this paper adopts a threshold-based image segmentation method. Due to the fact that the foreground and background of the processed image do not belong to a single-pixel range, it is necessary to achieve multi-threshold image segmentation. The paper adopts a segmentation method similar to the OTSU method with multiple thresholds while integrating Prewitt's threshold selection algorithm based on bimodal histograms [43,44]. Compared to the bimodal histogram method, it expands the number of peaks in the image histogram and determines the number of peaks based on the actual pixel composition and segmentation needs of the image. In order to improve the processing speed, the extreme points of the image histogram curve are used to determine the image segmentation threshold, which meets the extremely high UPH requirements through threshold segmentation.

2. The Image Histogram

Most threshold segmentation algorithms utilize the valleys of the grayscale image histogram curve to select the segmentation threshold [45,46]. For grayscale images, the histogram counts the occurrence number of each pixel and reflects the function between the image's gray-scale statistical characteristics and the gray-scale. Due to its low computational cost, the image histogram has been widely used in image processing, particularly in threshold segmentation of gray-scale and histogram equalization to achieve image feature enhancement, etc. [47,48]. Histogram equalization is computationally efficient and fast, and it yields significant enhancement of image features, especially for human visual perception. The following Figures 1–3 show the original image, the image after histogram equalization, and the histograms before and after histogram equalization.

Figure 1. The original image.

Figure 2. Image after histogram equalization.

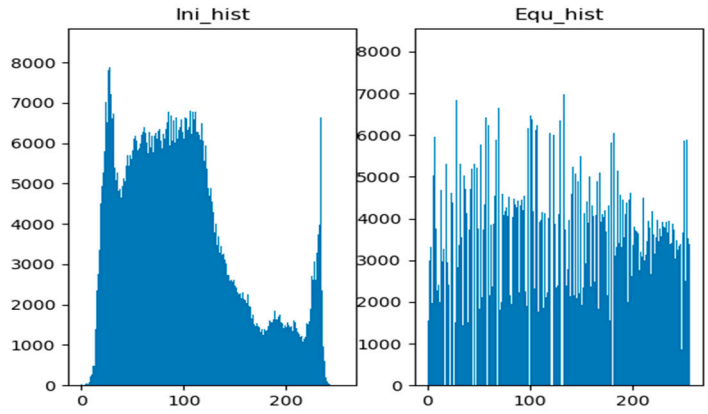

Figure 3. The histograms before and after histogram equalization.

The use of a histogram for image threshold segmentation has been widely applied, with the most typical approach being the OTSU method, also known as the maximum between-class variance method, which aims to maximize the variance between the foreground and background after segmentation. Most computer vision software currently integrates the OTSU algorithm API, such as the cv2.threshold() function parameter type in OpenCV, which uses cv2.THRESH_OTSU and can achieve image threshold segmentation by using the OTSU method. Figures 4–6 show the original image and the binarized image using the OTSU method, with a threshold segmentation value of 92.0 for binarization. The OTSU method can effectively segment ships, using them as the foreground and unifying the sky and seawater as the background. If it is necessary to segment the seawater or sky, the OTSU method will face significant difficulties and cannot simply use the sky or seawater as the foreground or background. In this case, a multi-threshold segmentation method needs to be used.

Figure 4. The original image.

Figure 5. The foreground image after binary segmentation using the OTSU method.

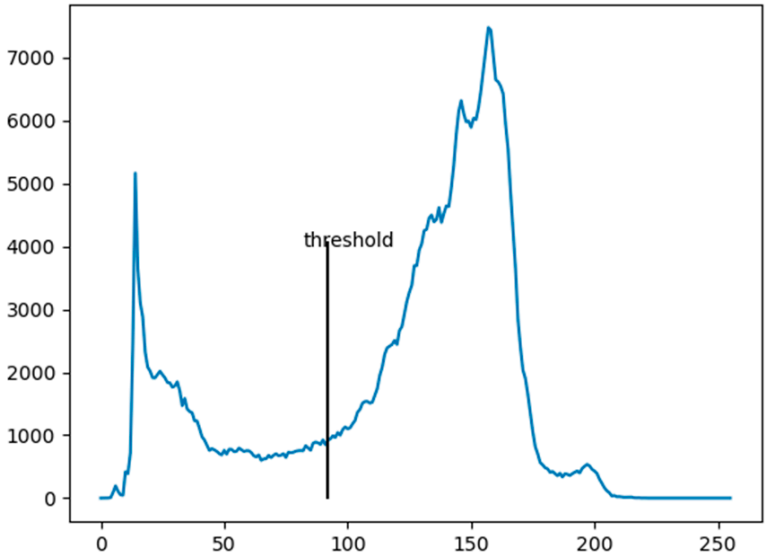

Figure 6. Threshold point in histograms using the OTSU method.

3. Threshold Segmentation Based on the OTSU Algorithm

3.1. The OTSU Algorithm

In image processing, it is common to segment an image into its foreground and background. Threshold segmentation is the binarization of the image, so it is necessary to determine the threshold segmentation point. Currently, the OTSU algorithm is a widely used method for determining the optimal threshold. The OTSU algorithm determines the optimal threshold by maximizing the between-class variance. Based on the grayscale characteristics of the image, the image is divided into background and foreground regions. The segmentation with the maximum between-class variance means the minimal misclassification probability.

Denoting the threshold for foreground and background segmentation as threshold, the ratio of background pixels to the total pixels number as ω_0, the average grayscale value of background pixels as μ_0, the ratio of foreground pixels to the total pixels number as ω_1, the average grayscale value of foreground pixels as μ_1, and the overall average grayscale value of the image as μ, the between-class variance is calculated as $\sigma^2{}_{max}$. The OTSU algorithm iterates through each pixel value $\in$ [0,255] to find the optimal segmentation threshold that maximizes the between-class variance, as shown in the following formula [49,50]. The algorithm principle of this method is simple, and all types of CV software have mature APIs that can be called, which has been widely applied.

$$\sigma^2{}_{max} = \omega_0 \times (\mu_0 - \mu)^2 + \omega_1 \times (\mu_1 - \mu)^2 \quad (1)$$

Among which $\omega_0 + \omega_1 = 1$, $\omega_0 \times \mu_0 + \omega_1 \times \mu_1 = \mu$.

3.2. Multi-Threshold Segmentation Based on the OTSU Algorithm

For some images, the foreground contains multiple grayscale ranges, such as cats, dogs, chickens, and ducks, in the same image, and each animal needs to be segmented. If we want to segment a specific foreground, such as dogs, we need to perform multiple segmentations based on different foreground grayscale values, which is referred to as multi-threshold segmentation. The OTSU algorithm can also be applied to multi-threshold image segmentation. For example, if we want to segment an image into three regions, including a background and two foreground regions with different grayscale ranges, we only need to modify the inter-class variance of the two classes to accommodate three classes.

Assuming the image is composed of background 0, foreground 1, and foreground 2, the proportion of pixels belonging to the background in the entire image pixels is denoted as ω_0, the average grayscale value of background pixels is denoted as μ_0, the proportion

of pixels belonging to foreground 1 is denoted as ω_1, the average grayscale value of foreground 1 pixels is denoted as μ_1, the proportion of pixels belonging to foreground 2 is denoted as ω_2, the average grayscale value of foreground 2 pixels is denoted as μ_2, the overall average grayscale value of the image is denoted as μ, and the inter-class variance is denoted as $\sigma^2{}_{max}$. We then traverse each pixel value $\in$ [0,255] to find the two values that maximize the inter-class variance, which corresponds to the optimal segmentation thresholds for foreground 1 and foreground 2, as shown in the following formula [51,52].

$$\sigma^2{}_{max} = \omega_0 \times (\mu_0 - \mu)^2 + \omega_1 \times (\mu_1 - \mu)^2 + \omega_2 \times (\mu_2 - \mu)^2 \quad (2)$$

Among which $\omega_0 + \omega_1 + \omega_2 = 1$, $\omega_0 \times \mu_0 + \omega_1 \times \mu_1 + \omega_2 \times \mu_2 = \mu$.

3.3. Limitations of OTSU Algorithm for Multi-Threshold Segmentation

In most industrial fields, threshold segmentation is just the first step in image processing to segment the parts of interest and, subsequently, achieve more complex processing, thus requiring fast image segmentation. The principle of the OTSU algorithm is clear and simple, and it is easy to implement in practical applications, but there are multiple shortcomings. Firstly, the OTSU algorithm involves the difference processing of each image pixel and exhaustive calculations for each grayscale value, resulting in relatively large computational complexity. When the image is large, especially when the multi-threshold segmentation is adopted, it will take a longer time. Secondly, the OTSU algorithm requires a significant proportion of the image to be occupied by the foreground to accurately calculate the segmentation threshold. Thirdly, the OTSU algorithm only considers the grayscale values of each pixel in segmentation, without considering its spatial distribution. When the number of grayscale levels to be segmented increases, the rationality of the segmentation decreases. Although many researchers have proposed improvements and optimizations to address the limitations of the OTSU algorithm, such as the two-dimensional OTSU algorithm [53,54]. However, the OTSU algorithm has a large computational complexity and time-consuming nature, which does not meet the requirements of fast segmentation, and still restricts its wider application [55,56].

The X-ray image of the underfill under white LED is shown in Figure 7. Each circle represents an LED chip, and the image covers a total of 15 LED chips in a 3 × 5 arrangement. The objective is to detect the adhesion of the underfill in each chip, specifically detect whether the middle solder pad (the square part in the middle) of the chip is missing, and detect the void rate of the solder pad. Since the relative positions of the chips in the image are not fixed during X-ray photography, and some chips may be partially outside the field of view, only 3 × 4 LED chips are visible in the image. Therefore, it is not possible to quickly identify the positions of the solder pads based on pixel locations. The segmentation of each solder pad image is required to evaluate the adhesion. The single image size is 1452 × 1000 pixels, with a 24-bit depth field for a single channel, and the average grayscale of the image in Figure 7 is 159.7. If the OTSU algorithm is used for multi-threshold segmentation, the image segmentation would take about 0.5 s per image. Considering image acquisition, transmission, and post-processing, the overall throughput UPH would not exceed 36 K, which does not match the production line speed.

Since the accuracy requirement for image segmentation is not high, the boundary of the solder pad is a regular rectangle, and the boundary can be accurately determined using other methods. The main function of threshold segmentation is to determine the presence and relative position of the solder pad. In order to achieve fast image segmentation with multiple thresholds and quickly identify the positions of solder pads, this paper proposes using the extremum points of the histogram curve as the segmentation threshold to reduce the computational complexity and achieve fast segmentation and high UPH.

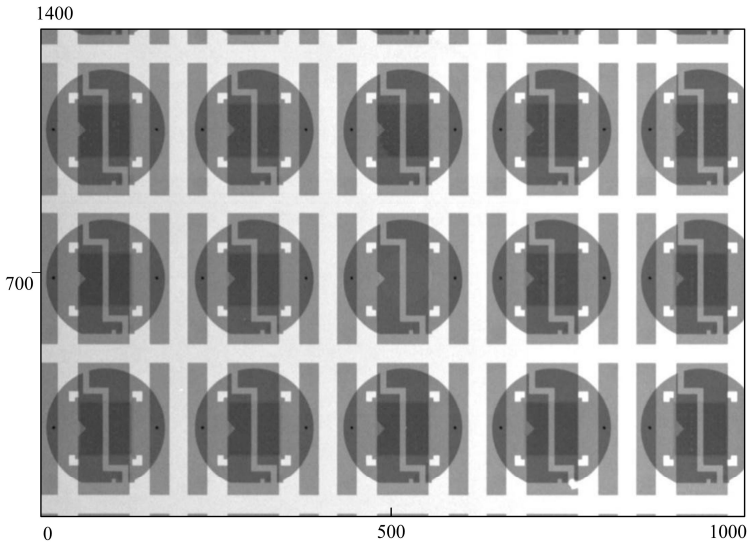

Figure 7. Adhesion of underfill in LED chips.

4. Fast Segmentation Based on the Histogram Technology

4.1. Histogram Curve

This paper proposes using the extremum points of the histogram curve to achieve quick image segmentation with multiple thresholds. Firstly, the histogram curve of the image depicting the adhesion of underfill in LED chips is obtained, as shown in Figure 8. The histogram curve exhibits multiple peaks and valleys. Based on the principles of histogram analysis, these peaks correspond to different grayscale value ranges in the original image. Each peak represents an image component within a specific grayscale range. If the image is divided into multiple grayscale ranges based on multiple peaks of the histogram curve, image segmentation can be achieved based on the grayscale range. Theoretically, the peak closest to the grayscale value of 0 in the histogram curve corresponds to the darkest regions in the original image. As shown in Figure 7, the corresponding part of the solder pads (the rectangular regions) are the darkest regions. By segmenting the image based on the first peak of the histogram curve, the solder pads can be extracted.

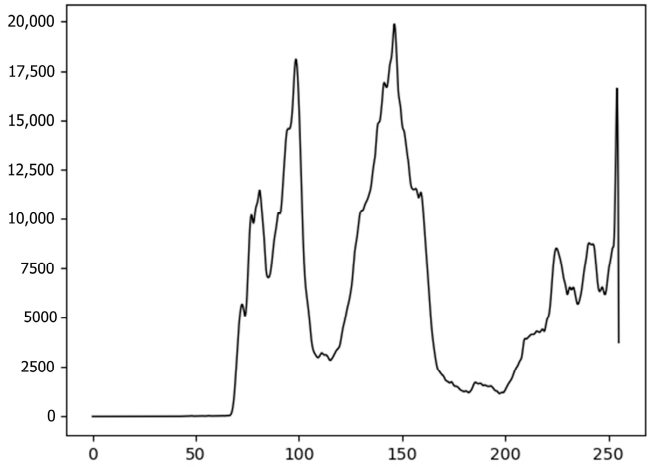

Figure 8. Histogram of the adhesion of the underfill in LED chips.

The histogram curve contains multiple peaks, and to separate each peak, the positions of the valleys in the curve are determined, which represent the minimum points. These minimum points serve as thresholds for image segmentation. By using each adjacent two minimum points, the corresponding peaks can be identified. Different combina-

tions of minimum points can be selected based on specific requirements to segment the desired parts.

4.2. Threshold Segmentation Based on the Minimum Point of the Histogram Curve

In order to find the extremum points of the histogram curve, an average filter is first applied to perform a certain degree of convolutional smoothing on the histogram curve to remove some burrs. As shown in Figure 9, the blue curve represents the original histogram curve, while the red curve represents the smoothed curve, and a total of seven minimum points are identified from left to right, while the small peak between the fourth and fifth minimum points contains very few image components. Peaks that meet a certain threshold can be set as true peaks, one can be eliminated at either the fourth or the fifth minimum point, or the sixth or seventh minimum point, which do not meet the criteria for true peaks. Therefore, there are three effective segmentation extreme points in Figure 9, which are the second, third, and fifth extreme points, corresponding to four peaks, and a total of four grayscale components in the image. Alternatively, a low-pass filter can be applied repeatedly until the desired number of peaks remains in the histogram, and then the threshold can be determined using the extremum points. In practical processing, the reasonable number of peaks in the histogram can be determined based on the actual situation. Small peaks and valleys can be filtered out by applying appropriate filtering or setting judgment conditions to select valuable minimum points. Finally, based on the corresponding peaks associated with the desired segmentation parts, the segmentation threshold can be selected from the retained minimum points to achieve fast multi-threshold segmentation of the image.

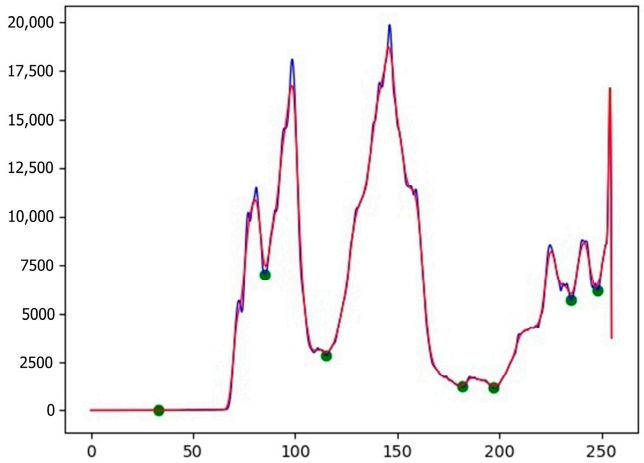

Figure 9. Illustration of extremum points in the histogram curve. The blue curve represents the original histogram curve, while the red curve represents the smoothed curve.

In the entire image, the LED solder pad is the darkest region with a gray value closest to 0. Therefore, this part belongs to the peak component with a gray value closest to 0 in the image histogram. Thus, the first and second minimum points are selected as the segmentation thresholds for the solder pad. The portion of the grayscale value in the image segmented using the threshold segmentation method between extremum point 1 and extremum point 2 happens to be the main body and some noise points of the LED pad in the image, as shown in Figure 10. The segmented image can be used to quickly determine the presence of 3×5 solder pads, among which the pads in the second row and third column are missing. At the same time, the approximate position of each pad can be quickly determined, providing support for subsequent precise determination of the solder pad boundaries. On a normal PC (Intel Core i5-8250U CPU, 8G RAM), all program codes for image segmentation and processing in the paper are written in Python language. And the total time for image adaptive finding extreme points and split pads is 0.0129 s, which is

significantly lower than the time required for the multi-threshold segmentation based on the OTSU algorithm.

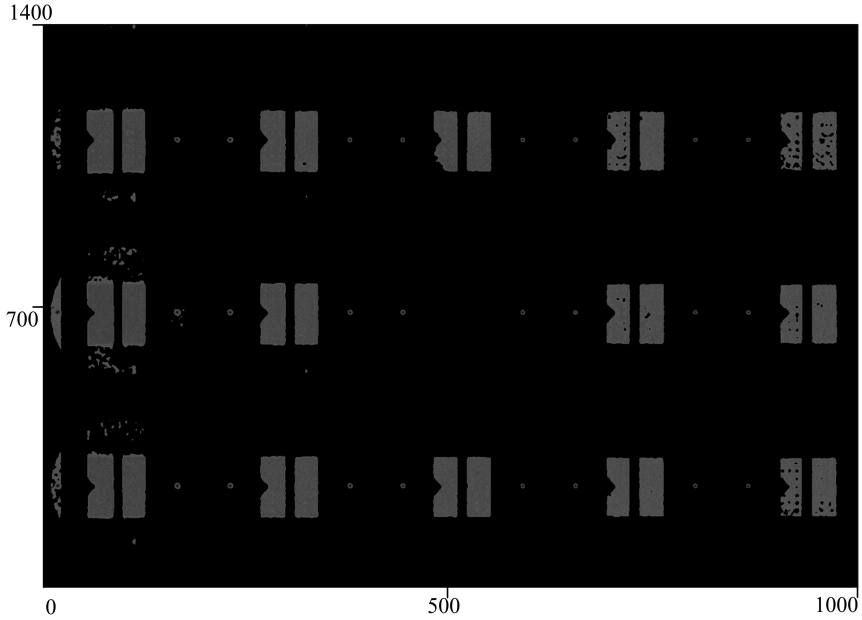

Figure 10. Solder pad images obtained by fast threshold segmentation based on the extremum points of the histogram curve.

4.3. Result of Threshold Segmentation Based on the Minimum Point of the Histogram Curve

After determining the missing situation and approximate position of the solder pads, the subsequent image processing involves accurately determining the solder pad boundaries and calculating the adhesion of the underfill in the solder pad. As shown in Figure 11, the final calculation and determination outcomes represent the adhesion of the solder pad. This method achieves fast segmentation of solder pads with processing accuracy comparable to the multi-threshold segmentation based on the OTSU algorithm and lower than the region growing method. The entire processing process, including the subsequent calculation of the adhesion of underfill in solder pads, can achieve a UPH of over 70 K, significantly improving efficiency.

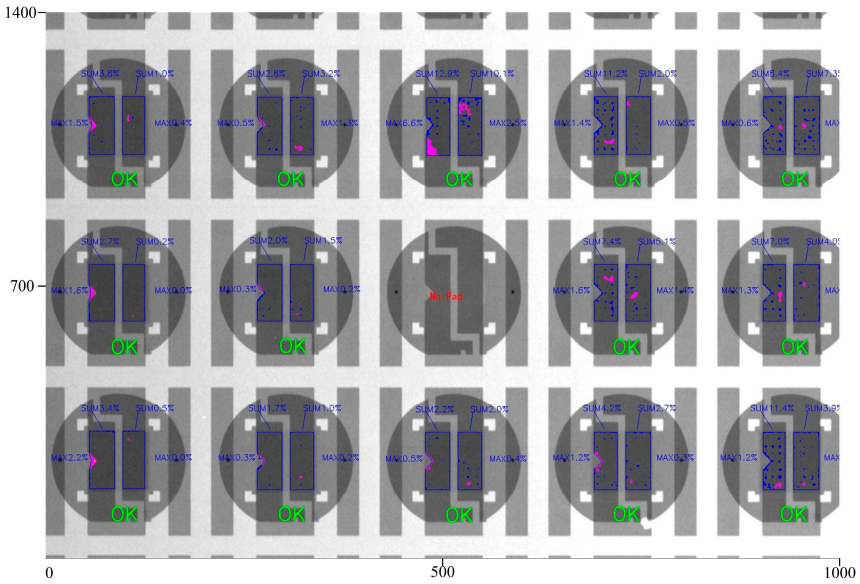

Figure 11. Final processing results of the adhesion of the underfill in LED chips.

In this example, the images of the underfill in white LED chips under X-ray belong to grayscale images with clear boundaries. In this paper, the multi-threshold segmentation based on OTSU and the region growing method are respectively adopted for image segmentation.

4.4. Result of Multi-Threshold Segmentation Based on OTSU

In this example, the image cannot be split out of the pad using the OTSU single threshold method, and the result of using the OTSU double threshold processing is shown in Figure 12. The image is split into 3 pixel components, and it takes 2.936 s to process a single picture on the same PC, and the OTSU double-threshold segmentation method takes hundreds of times longer than the histogram multi-threshold segmentation. It can be seen from the results that the double threshold method cannot accurately determine the presence or absence of pads, and according to the grayscale pixel composition of the image in Figure 9, at least the three-threshold segmentation method is required to accurately segment the pad image. If more threshold segmentation is used, the complexity and processing time of the algorithm will be greatly improved, the processing efficiency will be extremely low, and the value will be lost in industrial applications and UPH requirements will not be met.

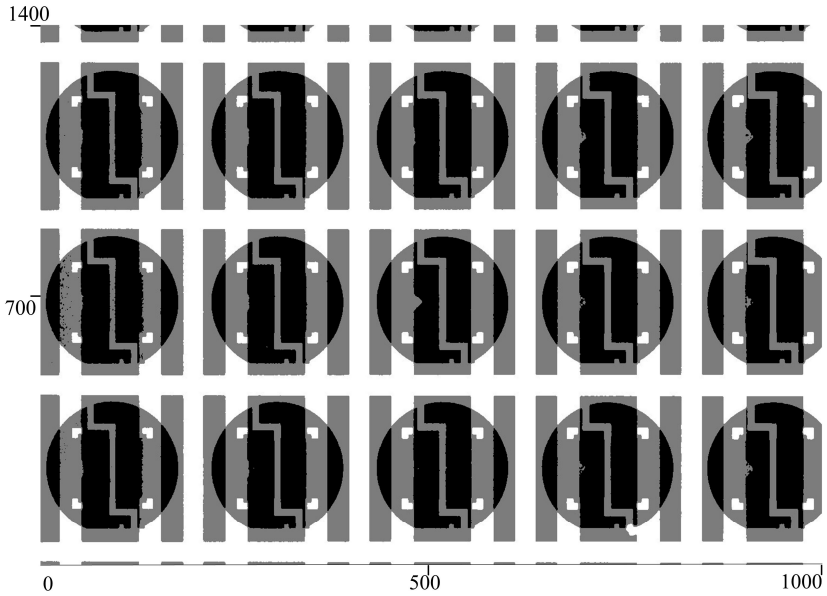

Figure 12. Processing results using OTSU double thresholds.

4.5. Result of the Region Growing Segmentation Method

In this example, the LED pads in the image are the darkest color parts. Multiple LED pads are distributed in one image, and the pad images with a good adhesion effect have similar overall grayscale values, belonging to the connected area with similar features. Therefore, it is suitable to use the region growing method for pad image segmentation.

In theory, if the number of pads is known, the number of seed points can be determined. As shown in Figure 7, a solder pad includes two parts of the PN electrode. There are a total of 30 pad areas in the figure, and the number of seed points is set to 30. The termination condition for region growing is the maximum area pixel value. However, when the seed point is set to 30, the actual segmentation effect is not good because there is a missing pad in the image, and there are areas around the non-fitting area that are similar to the pixels of the fitting area, as well as some incomplete pads. Therefore, we would consider increasing the seed point to 36, and the termination condition for region growing is the maximum region area of 10,800 pixels. This algorithm has the best segmentation effect, and processing a single image on the same PC takes 9.016 s, as shown in Figure 13.

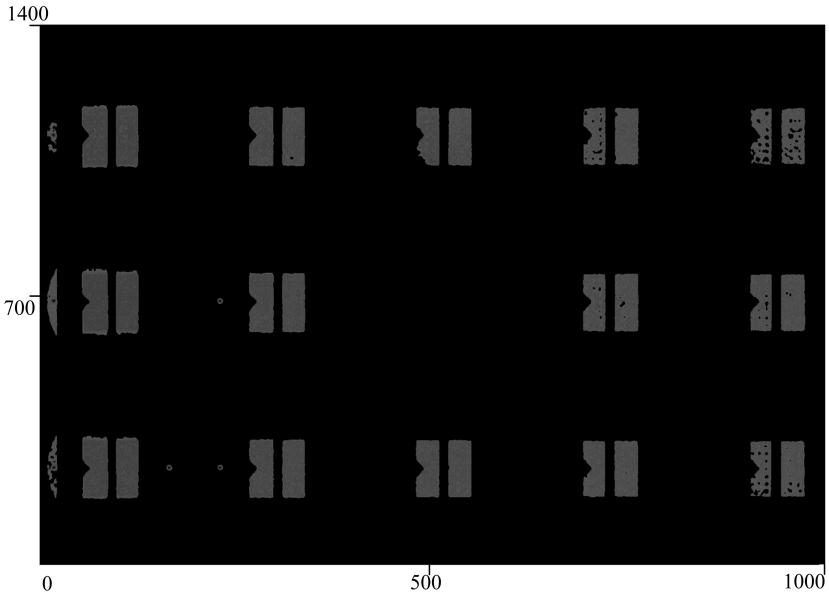

Figure 13. Processing results using the region growing method.

5. Discussion

5.1. Comparison of Different Segmentation Algorithms

As analyzed earlier, both the OTSU multi-threshold segmentation method and the region growing method have achieved a good segmentation effect on solder pads. In particular, the region growing method has the best segmentation effect, but it belongs to iterative algorithms. The more seed points, the greater the computational complexity, and the processing time required for a single image exceeds 9 s, which is significantly lower than the UPH requirements for industrial production. The segmentation effect of the OTSU algorithm meets the requirements. The OTSU double thresholds segmentation of a single image takes 2.936 s, and the comprehensive UPH is less than 1 K, which still does not meet the UPH requirements of industrial production. The segmentation method proposed in this article utilizes histogram curves to achieve a multi-threshold segmentation of images, with a UPH of over 70 K, which greatly facilitates the improvement of industrial production efficiency.

For the region growing algorithms, if too many seed points are set, there is a risk of excessive segmentation, and the calculation time is the longest, which is three times longer than the OTSU double-thresholds method. Although the segmentation accuracy is high, the comprehensive UPH will be less than 400, which is unacceptable for industrial production. In addition, if there is a significant difference in the overall grayscale value of images collected by different working conditions or X-ray machines on the industrial production line, the seed points need to be manually set. Table 1 lists the characteristics of the four segmentation methods proposed in this paper.

Table 1. Characteristics of different image segmentation methods.

No.	Methods	Segmentation Time	UPH	Segmentation Accuracy
1	the OTSU double threshold	2.936 s	<1 K	poor
2	Multi-threshold segmentation based on OTSU	>2.936 s	<1 k	good
3	Region growing segmentation method	9.016 s	<0.4 K	best
4	Threshold segmentation based on the minimum point of the histogram curve	0.0129 s	≥70 K	good

5.2. Adaptability of Images with Different Grayscale Values

When the overall grayscale values of the images are different between different working conditions or different X-ray machines on the industrial production lines, the threshold segmentation based on the minimum point of the histogram curve can still adapt to the change of the grayscale value and quickly realize the multi-thresholds segmentation of the image. As shown in Figure 14, the overall average grayscale value of the image is 226.5, while the overall grayscale value of Figure 7 is 159.7. Figure 15 is an image processed by using the histogram curve extremum method, which determines the segmentation threshold according to the image histogram and does not need to predict the segmentation range of foreground pixels and background in advance, which can realize the adaptive segmentation.

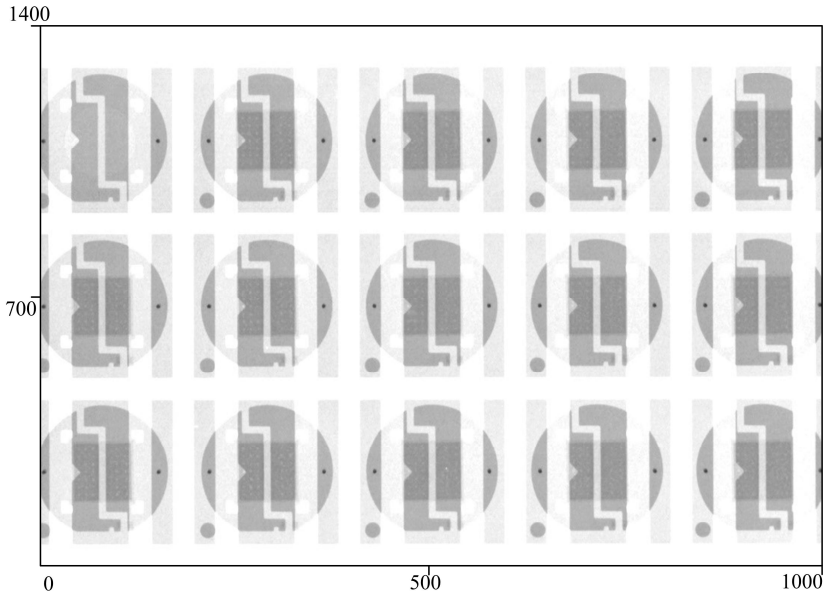

Figure 14. Original image with an average grayscale value of 226.5.

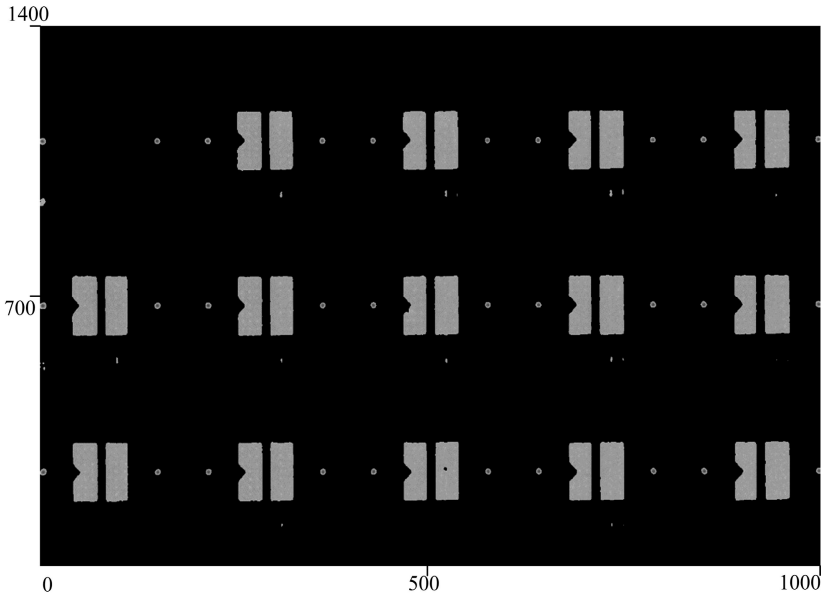

Figure 15. Image processing result with an average grayscale value of 226.5.

6. Conclusions

In this article, the OTSU multi-threshold segmentation method and the region growing method were used for the solder pad image segmentation of the adhesion of underfill in

LED chips. Both methods achieved a good segmentation effect of solder pads, but the time consumption of both methods did not meet the UPH requirements of industrial production.

In response to the high computational complexity and slow processing speed of the OTSU multi-threshold segmentation method and the region growing method when processing large images, this paper proposes the use of extreme points on the histogram curve as segmentation thresholds to achieve a multi-threshold segmentation of images. The processing accuracy of this method meets most of the requirements of image processing, and the processing speed is significantly improved. In the segmentation process of solder pads for the cohesion of underfill in LED chips, the time consumption is less than 1/100 of the OTSU multi-threshold algorithm, which significantly improves the efficiency of image post-processing. The entire processing process has a UPH of up to 70 K and has strong adaptability for images with significant differences in grayscale values, which better meets the comprehensive requirements of achieving the fastest efficiency with appropriate accuracy in industrial production.

Although the OTSU algorithm, the clustering algorithm, and the region growing algorithm have all achieved significant results in some image processing fields, when the segmentation threshold is greater than two, the processing speed significantly decreases. The method proposed in this article has significant advantages in processing images with clear boundaries, or when the background is not smooth or the accuracy requirements for image segmentation are not high. Especially for multi-threshold segmentation, it can quickly achieve image multi-threshold segmentation with low computational power requirements. For example, in industrial production, when processing images of metal product processing surfaces, this method can quickly segment products. When the image boundary is unclear or the foreground area is not coherent enough in the grayscale distribution, it can be understood that there are no obvious minimum points at the expected segmentation points of the histogram curve, and the two peaks of the curve are relatively horizontal. This method may not achieve accurate segmentation.

Author Contributions: Conceptualization, M.X.; methodology, X.G.; software, S.C.; validation, Q.Y.; formal analysis, X.L.; investigation, Y.K.; resources, S.C.; writing—original draft preparation, M.X.; writing—review and editing, C.H.; supervision, X.G.; funding acquisition, M.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Independent Scientific Research Project of the University (No. 2022505070).

Data Availability Statement: The data can be shared upon request.

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

References

1. Borsotti, M.; Campadelli, P.; Schettini, R. Quantitative evaluation of color image segmentation results. *Pattern Recognit. Lett.* **1998**, *19*, 741–747. [CrossRef]
2. Cardoso, J.S.; Corte-Real, L. Toward a generic evaluation of image segmentation. *IEEE Trans. Image Process.* **2005**, *11*, 1773–1782. [CrossRef] [PubMed]
3. Ciesielski, K.C.; Udupa, J.K. A framework for comparing different image segmentation methods and its use in studying equivalences between level set and fuzzy connectedness frame-works. *Comput. Vis. Image Underst.* **2011**, *115*, 721–734. [CrossRef] [PubMed]
4. Crevier, D. Image segmentation algorithm development using ground truth image data sets. *Comput. Vis. Image Underst.* **2008**, *112*, 143–159. [CrossRef]
5. Erdem, C.E.; Sankur, B.; Tekalp, A.M. Performance measures for video object segmentation and tracking. *IEEE Trans. Image Process.* **2004**, *13*, 937–951. [CrossRef] [PubMed]
6. Huang, P.; Zheng, Q.; Liang, C. Overview of Image Segmentation Methods. *J. Wuhan Univ. Nat. Sci. Ed.* **2020**, *66*, 519–531. [CrossRef]
7. Zhou, L.; Jiang, F. Survey on image segmentation methods. *Appl. Res. Comput.* **2017**, *34*, 1922–1928.
8. Zhang, Y.; Yuan, J.; Liu, H. Overview of Image Segmentation Algorithm. *Comput. Sci.* **2015**, *42*, 29–32.
9. Ma, R.; Zeng, W.; Song, G.; Yin, Q.; Xu, Z. Pythagorean fuzzy C-means algorithm for image segmentation. *Int. J. Intell. Syst.* **2021**, *36*, 1223–1243. [CrossRef]

10. Glasbey, C.A. An analysis of histogram based thresholding algorithm. *CVGIP Graph. Models Image Process.* **1993**, *55*, 532–537. [CrossRef]
11. Hashemi, S.E.; Gholian-Jouybari, F.; Hajiaghaei-Keshteli, M. A Fuzzy C-Means Algorithm for Optimizing Data Clustering. *Expert Syst. Appl.* **2023**, *227*, 120377. [CrossRef]
12. Chang, C.I.; Chen, K.; Wang, J.; Althouse, M.L. A relative entropy-based approach to image thresholding. *Pattern Recognit.* **1994**, *27*, 1275–1289. [CrossRef]
13. Ramon, L.C.; Varsheny, P.K. Image thresholding based on Ali-Silvey distance measures. *Pattern Recognit.* **1994**, *30*, 1161–1174.
14. Sahoo, P.K.; Wilkings, C.; Yeages, J. Threshold selection using Renyi's entropy. *Pattern Recognit.* **1994**, *30*, 71–84. [CrossRef]
15. Lewng, C.K.; Lam, F.K. Maximum segmental image information thresholding. *CVGIP Graph. Models Image Process.* **1998**, *60*, 57–76. [CrossRef]
16. Ge, F.; Wang, S.; Liu, T. New benchmark for image segmentation evaluation. *J. Electron. Imaging* **2007**, *16*, 033011.
17. Hao, J.; Shen, Y.; Xu, H.; Zou, J. A Region Entropy Based Objective Evaluation Method for Image Segmentation. In Proceedings of the IEEE International Conference on Instrumentation and Measurement Technology, Singapore, 5–7 May 2009; pp. 373–377.
18. Guo, Q.; Wang, Y.; Yang, S.; Xiang, Z. A method of blasted rock image segmentation based on improved watershed algorithm. *Sci. Rep.* **2022**, *12*, 7143. [CrossRef]
19. Gao, X.; Zhang, Y.; Wang, H.; Sun, Y.; Zhao, F.; Zhang, X. A modified fuzzy clustering algorithm based on dynamic relatedness model for image segmentation. *Vis. Comput. Int. J. Comput. Graph.* **2023**, *39*, 1583–1596. [CrossRef]
20. Xu, G.; Feng, C.; Ma, F. Review of Medical Image Segmentation Based on UNet. *J. Front. Comput. Sci. Technol.* **2023**, *17*, 1776–1792.
21. Hrdina, J.; Matoušek, R.; Tichý, R. *Scopus Preview—Colour Image Segmentation by Region Growing Based on Conformal Geometric Algebra (Conference Paper)*; Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer International Publishing: Berlin/Heidelberg, Germany, 2019; Volume 11542, pp. 564–570.
22. Zhou, J.; Yang, M. Bone Region Segmentation in Medical Images Based on Improved Watershed Algorithm. *Comput. Intell. Neurosci.* **2022**, *2022*, 3975853. [CrossRef]
23. Jiang, F.; Gu, Q.; Hao, H.Z.; Li, N.; Guo, Y.W.; Chen, D.X. Survey on content-based image segmentation methods. *Ruan Jian Xue Bao/J. Softw.* **2017**, *28*, 160–183. (In Chinese)
24. Rajakani, K.; Abdulsahib, A.K.; Kamaruddin, S.S.; Jabar, M.M. A Double Clustering Approach for Color Image Segmentation. *Wirel. Commun. Mob. Comput.* **2023**, *2023*, 1039870.
25. Xing, Y.; Zhong, L.; Zhong, X. Study of Clustering Algorithms in Object Tracking and Image Segmentation. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 1530–8669. [CrossRef]
26. Deeparani, K.; Sudhakar, P. Efficient image segmentation and implementation of K-means clustering. *Mater. Today Proc.* **2021**, *45*, 8076–8079. [CrossRef]
27. Li, T.; Xu, Y.; Luo, J.; He, J.; Lin, S. Region-Growing Algorithm on CT Angiography Images for Detection of Gynecological Malignant Tumor. *Sci. Program.* **2021**, *2021*, 1–7.
28. Raja, J.A.; Babu, N.K. Adaptive Region Growing Image Segmentation Algorithms for Breast MRI. *Int. J. Recent Technol. Eng.* **2019**, *8*, 8729–8732. [CrossRef]
29. Haralick, R.; Shapiro, L.G. Survey: Image segmentation techniques. *CVGIP Graph. Models Image Process.* **1985**, *29*, 100–132. [CrossRef]
30. Hoover, A.; Jean-Baptiste, G.; Jiang, X.; Flynn, P.J.; Bunke, H.; Goldgof, D.B.; Fisher, R.B. An experimental comparison of range segmentation algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.* **1996**, *18*, 673–689. [CrossRef]
31. Kadapala, B.K.R.; Hakeem, K.A. Region Growing based Automatic Localized Adaptive Thresholding Algorithm for Water Extraction using Sentinel-2 MSI Imagery. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1. [CrossRef]
32. Lee, S.U.; Chung, S.Y.; Park, R.H. A comparative performance study of several global thresholding techniques for segmentation. *Comput. Vis. Image Underst.* **1990**, *52*, 171–190. [CrossRef]
33. Lei, T.; Udupa, J.K. Performance evaluation of finite normal model-based image segmentation technique. *IEEE Trans. Image Process.* **2003**, *12*, 1163–1169.
34. Martin, A.; Laanaya, H.; Arnold-Bos, A. Evaluation for uncertain image classification and segmentation. *Pattern Recognit.* **2006**, *39*, 1987–1995. [CrossRef]
35. Ortiz, A.; Oliver, G. On the use of overlapping area matrix for image segmentation evaluation: A survey and new performance measures. *Pattern Recognit. Lett.* **2006**, *27*, 1916–1926. [CrossRef]
36. Peng, R.; Varshney, P.K. On performance limits of image segmentation algorithms. *Comput. Vis. Image Underst.* **2015**, *132*, 24–38. [CrossRef]
37. Unnikrishnan, R.; Pantofaru, C.; Hebert, M. Toward objective evaluation of image segmentation algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 929–944. [CrossRef]
38. Mazurowski, M.A.; Dong, H.; Gu, H.; Yang, J.; Konz, N.; Zhang, Y. Segment anything model for medical image analysis: An experimental study. *Med. Image Anal.* **2023**, *89*, 102918. [CrossRef]
39. Hu, C.; Li, X. When SAM Meets Medical Images: An Investigation of Segment Anything Model (SAM) on Multi-phase Liver Tumor Segmentation. *arXiv* **2023**, arXiv:2304.08506.

40. Friebel, A.; Johann, T.; Drasdo, D.; Hoehme, S. Guided interactive image segmentation using machine learning and color-based image set clustering. *Bioinformatics* **2022**, *38*, btac547. [CrossRef]
41. Vojodi, H.; Fakhari, A.; Moghadam AM, E. A new evaluation measure for color image segmentation based on genetic programming approach. *Image Vis. Comput.* **2013**, *31*, 877–886. [CrossRef]
42. Zhang, H.; Fritts, J.E.; Goldman, S.A. Image segmentation evaluation: A survey of unsupervised methods. *Comput. Vis. Image Underst.* **2008**, *110*, 260–280. [CrossRef]
43. Ju, Z.; Xue, Y.; Zhang, W.; Zhai, C. Algorithm for detecting pomegranate disease spots based on Prewitt operator with adaptive threshold. Transactions of the Chinese Society of Agricultural Engineering. *Trans. Chin. Soc. Agric. Eng.* **2020**, *36*, 135–142.
44. Prakash, N.; Basha, S.A.; Chowdhury, S.; Reshmi, B.; Kapila, D.; Devi, S. Implementation of Image Segmentation with Prewitt Edge Detection using VLSI Technique. In Proceedings of the 2022 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), Chennai, India, 15–16 July 2022.
45. Min, L.; Wang, H.; Jiao, J. A Review of the Optical Remote Sensing Image Segmentation Technology. *Spacecr. Recovery Remote Sens.* **2020**, *41*, 1–13. (In Chinese)
46. Hsu, C.Y.; Shao, L.J.; Tseng, K.K.; Huang, W.T. Moon image segmentation with a new mixture histogram model. *Enterp. Inf. Syst.* **2021**, *15*, 1046–1069. [CrossRef]
47. Li, M.; Wang, L.; Deng, S.; Zhou, C. Color image segmentation using adaptive hierarchical-histogram thresholding. *PLoS ONE* **2020**, *15*, e0226345. [CrossRef] [PubMed]
48. Naderi Boldaji, M.R.; Hosseini Semnani, S. Color image segmentation using multi-objective swarm optimizer and multi-level histogram thresholding. *Multimed. Tools Appl.* **2022**, *81*, 30647–30661. [CrossRef]
49. Li, Y.K.; Yang, S.W.; Liu, T. An automatic threshold selection approach for remote sensing imagery of multimodal histograms. *J. Lanzhou Jiaotong Univ.* **2013**, *32*, 199–204.
50. Liu, G.; Zhang, Z.; Cui, X.; Kuang, J.; Cai, J.; Ji, X. Chromosome Image Segmentation Based on OTSU and Region Growing Algorithm. In Proceedings of the 2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), Chengdu, China, 19–21 August 2022.
51. Suryani, E.; Asmari, E.I.; Harjito, B. Image Segmentation of Acute Myeloid Leukemia Using Multi Otsu Thresholding. *J. Phys. Conf. Ser.* **2021**, *1803*, 012016. [CrossRef]
52. Li, H.; Yao, L.; Shi, L. Automatic selection of image threshold based on improved Otsu. *Comput. Simul.* **2007**, *24*, 216–220.
53. Xu, X.Y.; Song, E.M.; Jin, L.H. Characteristic analysis of threshold based on Otsu criterion. *Acta Electron. Sin.* **2009**, *37*, 2716–2719.
54. Qu, K.; Zheng, L. Automatic thresholding of gray-scale image based on the proportion of object and background. *Appl. Sci. Technol.* **2010**, *37*, 52–54.
55. Wu, B.; Zhou, J.; Ji, X.; Yin, Y.; Shen, X. An ameliorated teaching–learning-based optimization algorithm based study of image segmentation for multilevel thresholding using Kapur's entropy and Otsu's between class variance. *Inf. Sci.* **2020**, *533*, 72–107. [CrossRef]
56. Zheng, J.; Gao, Y.; Zhang, H.; Lei, Y.; Zhang, J. OTSU Multi-Threshold Image Segmentation Based on Improved Particle Swarm Algorithm. *Appl. Sci.* **2022**, *12*, 11514. [CrossRef]