



An improved K-means algorithm for underwater image background segmentation

Wei Chen^{1,2} · Cenyu He¹ · Chunlin Ji³ · Meiyong Zhang² · Siyu Chen⁴

Received: 15 July 2020 / Revised: 8 October 2020 / Accepted: 10 February 2021 /

Published online: 12 March 2021

© The Author(s) 2021

Abstract

Conventional algorithms fail to obtain satisfactory background segmentation results for underwater images. In this study, an improved K-means algorithm was developed for underwater image background segmentation to address the issue of improper K value determination and minimize the impact of initial centroid position of grayscale image during the gray level quantization of the conventional K-means algorithm. A total of 100 underwater images taken by an underwater robot were sampled to test the aforementioned algorithm in respect of background segmentation validity and time cost. The K value and initial centroid position of grayscale image were optimized. The results were compared to the other three existing algorithms, including the conventional K-means algorithm, the improved Otsu algorithm, and the Canny operator edge extraction method. The experimental results showed that the improved K-means underwater background segmentation algorithm could effectively segment the background of underwater images with a low color cast, low contrast, and blurred edges. Although its cost in time was higher than that of the other three algorithms, it none the less proved more efficient than the time-consuming manual segmentation method. The algorithm proposed in this paper could potentially be used in underwater environments for underwater background segmentation.

Keywords Underwater image · Image segmentation · Background segmentation · K-means algorithm

1 Introduction

The significance of ocean exploration has been highlighted by researchers [19]. Within the field of ocean exploration, underwater detection technology for identifying objects in underwater environments is of high significance from a number of security and recovery perspectives, and has become a topic for extensive research. Although many image segmentation

✉ Siyu Chen
chensy-nj@163.com

Extended author information available on the last page of the article

models have been developed to detect objects in various above-water contexts [11–16, 26], conventional electro-optical imaging methods perform poorly underwater, resulting in underwater detection and extraction proving a challenging predicament. Trained and experienced personnel are still required to interpret images obtained from echogram methodology such as sonar or ultrasound imaging. Even with training, distinguishing objects of interest from their background in an echogram can be difficult. [9]. It is this background impediment that triggered exploratory research in underwater image detection techniques.

According to the law of Physics, light is absorbed and dispersed when transmitted through water [9], thereby decreasing its energy and resulting in reduced observation distance for underwater light imaging [5]. It is well documented that most segmentation methods are highly susceptible to the impact of changing light conditions during image acquisition, as well as color variations of the background of the scene. Underwater images furthermore suffer from non-uniform brightness, poor contrast, diminished coloration, significant blur [25]. It is thus necessary to perform a preliminary treatment on those images prior to using image processing methods such as background segmentation. Background segmentation is a technique that aims to split digital images in such a way as to differentiating the features in the foreground from the background. The separate layers comprise homogeneous properties such as color, texture, and brightness [31]. To develop specific background segmentation algorithms suitable for application to underwater images affected by several drawbacks such as non-uniform brightness, poor contrast, and significant blur would therefore be of immense interest and benefit. Several background segmentation models such as histogram-threshold, edge-detection, and semantic algorithms have been developed within the relevant research community [1, 6, 7, 10, 15, 20–25, 29, 32, 35, 38].

Although histogram-threshold background segmentation models can easily be carried out based on gray values, they can only produce good segmentation results if the image has a high gray level contrast. Improved models such as the image-based statistical threshold method and the minimum error method have thus been developed [24]. In addition, the spatial structure of images is often ignored, and the segmentation model is highly sensitive to environmental noise, which results in low detection accuracy of objects in underwater images. Few attempts have been made to date to improve background segmentation results for underwater images [7, 25].

Segmentation techniques such as a semantic segmentation based on pixel scores have been developed [15, 29, 32, 38]. While conventional image segmentation methods hardly rely on spatial information within the target region, reducing image segmentation accuracy. Recently, semantic image segmentation models were developed using a feature fusing model [11, 13, 15]. Alternatively, the edge-detection background segmentation algorithm based on the local maximum of the image gradient can reflect the spatial details of the images that comprise sharp edges and little noise within the smoothing region of the image [30]. In cases where underwater images normally present blurred or discontinuous edges with increased noise, it is difficult to achieve a good segmentation result using the edge detection-based segmentation model.

In summary, owing to the blurring characteristics of underwater image edges, their low contrast brightness and low resolution, existing background segmentation algorithms fail to produce ideal results when applied to underground water images.

The K-means algorithm was recently applied to image segmentation due to its low algorithm complexity and simple implementation. When using this K-means algorithm for gray level quantization, the pixels between the background and foreground image within a specific region could be distinguished. Threshold segmentation could then be performed to remove the background within the said region. However, conventional K-means algorithm

results are strongly dependent on the choice of initial centroid. This is because the image segmentation threshold based on the K-means algorithm is only locally optimal, and the locally optimal threshold cannot equate the globally optimal threshold. The gray level quantization order obtained by the algorithm to find maximum in an array is furthermore set and extracted by users. As a background segmentation algorithm, the K-means algorithm can work well for dynamic thresholding and is suitable for above-water images. However, when dealing with the background segmentation of underwater images, the gray level of the background image pixels is similar to that of the foreground ones. Therefore, incorrect initial centroid setting of the algorithm, or highly limited quantization orders are more likely to divide the foreground image pixels into part of the background image pixels, resulting in an incomplete final segmentation result. Hence, it is clear that the negative influence of initial centroid setting on the background segmentation accuracy should be minimized for the successful application of underwater image segmentation models.

This study proposes an improved K-means algorithm in underwater image background segmentation. The method described in this paper used the Lab color space for image color correction to reduce interference from underwater color cast on image segmentation; a K-means algorithm was used to perform histogram analysis of gray image; the histogram of the quantized gray image was analyzed. 100 underwater images were selected to test the new algorithm. The results were evaluated and compared to those of manual background segmentation and other background segmentation algorithms, including the improved Otsu algorithm, the Canny operator edge extraction, and the conventional K-means algorithm. The new underwater image segmentation model optimized the combination of the finite contrast histogram equalization algorithm and the color correction algorithm based on equivalent circular color cast detection. The image contrast was enhanced and the interference of non-uniform brightness was reduced. The influence of image color cast was furthermore reduced. The new algorithm strategy addressed the issue of improper K value determination and minimized the impact of initial centroid position of grayscale image during the gray level quantization of the conventional K-means algorithm. It is proposed that the new model could offer new insights into the application of underwater background segmentation in underwater environments.

2 Related work

Three aspects of related studies are reviewed: background segmentation techniques based on histogram threshold, on edge detection, as well as on semantic image segmentation.

2.1 Background segmentation techniques based on histogram threshold

Background segmentation based on histogram threshold offered the benefits of simple and easy implementation, small calculation amount, and fast segmentation speed. A threshold value was set as the basis for image segmentation. The gray level of the pixel that exceeded the threshold was designated as the maximum gray level, while the opposite was the minimum gray level. This separation divided the image into several meaningful areas. The core of the algorithm challenge was how to determine the gray threshold, which is generally selected according to the gray histogram characteristics of the image. In cases where the gray level contrast of the image was high, the threshold segmentation algorithm enabled a good image segmentation effect to be obtained. Researchers proposed many improved algorithms for

threshold selection and segmentation, such as the image-based statistical threshold method and the minimum error method proposed by Kittler et al. [24], a two-dimensional entropy threshold method proposed by Abutaleb [1], an improved Otsu algorithm proposed by Hu et al. [21] and Du et al. [18], and a multi-threshold image segmentation based on chaotic particle swarm algorithm proposed by Jiang and Li [23].

However, the above algorithms present several limitations. In histogram threshold segmentation, the statistical characteristics of the gray level of the image are taken into account, but its spatial structure is often ignored, especially useful features such as the texture characteristics of the image. Moreover, histogram threshold segmentation is highly sensitive to environmental noise. The image divided by the histogram threshold was found potentially to increase salt and pepper noise, especially in cases of low brightness and low contrast. Li et al. [25] used the modified local entropy-based transition region extraction and thresholding to segment the underwater image. They found that this method offered a high capacity to suppress noise, but because the gray level between target and background were similar, adhesions between them were prone to occurring in the segmentation result. Cao et al. [7] proposed an image segmentation method based on the custom color space model (i.e., HSV model). The saturation *S* was taken as the main analysis channel to conduct the segmentation sequence on different color components. However, the *S* component was more affected by the illumination intensity and incident angle of the light. Due to the variable sensitivity of the surface of underwater imaging objects, successful complete separation of the target from the background is unlikely to be achieved. Therefore, in underwater environments, due to the influence of the water medium on light dispersion and absorption, water impurities, and underwater lighting conditions, the outcome of image segmentation proved less than ideal.

2.2 Semantic image segmentation

Semantic segmentation techniques that divided an image into regions according to various semantic data was also used to achieve image segmentation [15, 29, 32, 38]. Among these semantic segmentation models, candidate region-based models that first described and classified the free-form features of regions once the free-form regions had been extracted from the image were selected. These models then converted the region-based prediction to a pixel-level prediction according to the region with the highest pixel score to mark pixels. Similarly to histogram threshold segmentation models, this pixel-score based image segmentation technique failed to utilize the spatial information in the candidate region, impacting the image segmentation outcome. Although the semantic segmentation models based on the fully convolutional symmetric semantic segmentation model improved image segmentation results, calculation costs to obtain training samples with pixel-level labels were high, and sensitivity to object location was lost. Attempts were thus recently made to improve the above semantic image segmentation models by using a feature-fusing model with layer-by-layer context features [11, 13, 15].

2.3 Background segmentation techniques based on edge detection

Unlike histogram threshold segmentation, the background segmentation algorithm for edge detection is a segmentation processing algorithm based on image edge detection. The core of the image segmentation algorithm based on edge detection consists in finding the region comprising the local maximum of the image gradient and using this region as the basis for image segmentation. The edge information belongs to the high-frequency component of the

image, which reflects its structural information of the image, such as the contour shape of objects. The edge detection algorithm can obtain object edge information and depict the target object location, so that the computer can identify the target object. The edge-based segmentation algorithm comprises of edge detection operator methods such as gradient, Laplacian, and template operation operator [20]. For video segmentation, a dynamic programming method was introduced by Wu et al. [35]. Bo et al. [6] processed the edge-detected images by constructing a Sobel operator with eight directional templates combined with an iterative segmentation threshold algorithm and an omnidirectional expansion morphology method.

The background segmentation algorithm based on edge detection proved capable of achieving good results for images with sharp edges and little noise within the image smoothing region. However, when the image edges were blurred or there were greater high-frequency noises, the edge detection-based segmentation algorithm often achieved undesired outcomes, such as the erroneous removal of the foreground image due to the discontinuity of the blurred region border, or due to the high frequency of the image. Noise created a false area within the output image, that is, a resulting 'false' absence of border where a boundary actually existed, and a 'false' presence of an edge in an area where no border existed due to the noise.

Many studies used edge detection-based segmentation algorithms to process underwater imaging. Chen et al. [222] proposed an algorithm based on definite weighting coefficients entropy to determine borders. These algorithms were simple to implement, but their associated noise-reduction effect was not evident, their edge detection accuracy was not high, continuity was lacking, and most importantly they were not adaptable. Due to the harsh underwater imaging environment, images formed via natural illumination tended to be blurred at the edges. However, the addition of artificial fill light generated false borders due to non-uniform illumination. Therefore, this type of image segmentation algorithm frequently detected false edges or failed to detect local edges in the course of underwater image processing. Some border details were susceptible to being mistakenly interpreted as noise due to the similarity, and hence to being removed.

3 Research methods

The algorithm described in this paper lent itself to being divided into three parts: color adjustment, gray level quantization, and background judgment and segmentation. Due to the absorption and dissemination of ambient light as well as the impact of illumination sources on the underwater robot, the color of underwater images, which were dominated by shades of green or blue, could be modified. It was hence possible as a result to interfere with the identification and removal of background areas. [5]. It was thus necessary to perform a pre-treatment to correct the underwater image color. In this paper, the finite contrast histogram equalization algorithm proposed by Yang et al. [37] was used to improve image contrast and reduce interference from non-uniform brightness. The color correction algorithm based on equivalent circular color cast detection presented by Xu et al. [36] was utilized to reduce the influence of image color cast.

Gray level quantization generally uses a uniform quantization method [4], that is, where the quantization interval is uniform; this could approximate the gray shade of the foreground object region close with that of the background, preventing differentiation between foreground object and background. To avoid this, the K-means algorithm [17] was used to quantize distinct gray shades within images. In the present study, each gray level mask was produced to

obtain a color image for the location of each gray level, and the background image was determined according to the general characteristics (e.g. color, intensity) of the underwater image background, thereby removing the background from the underwater image. The full algorithm flow chart is shown in Fig. 1 below. Details of gray level quantization, background judgment, and background segmentation are described in Sections 2.1–2.2 hereafter.

3.1 Gray level quantization

3.1.1 K-means algorithm

It should be noted that K-mean clustering is a technique that groups n pixels of an image into K numbers of clusters, where $K < n$ and K is a positive integer [2]. The purpose of a clustering algorithm is to divide the data set made up of various elements into different data subsets, each of which contains at least one element, and each element in the data set belongs to a subset, but only one subset, and is clustered. The subset obtained by the class is also called a data cluster. The K-means algorithm performs clustering operations on known data sets without other trial data sets to assist clustering; the K-means algorithm is therefore an unsupervised clustering algorithm.

The K centroid is first defined by the user. After determining the starting position of the centroid, the data set data are divided into the data cluster nearest to the centroid location. There are several methods to measure distance, such as Euclidean, block, and cosine distance. The centroid of the data cluster is then calculated for each data cluster obtained, first by a

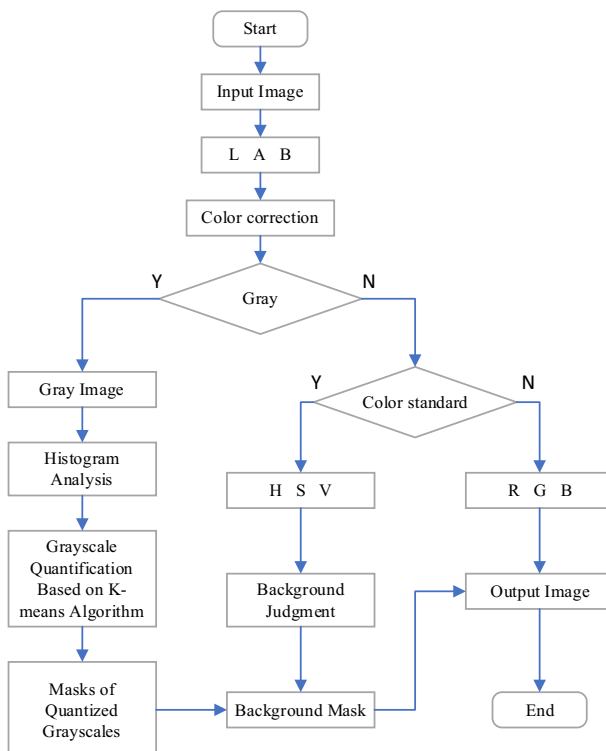


Fig. 1 Algorithm flow chart

division, and then as the updated value for each centroid. Data partitioning and centroid updating operations are repeated until the centroid position of the divided data cluster no longer changes, or when the distance change is less than a certain value. The algorithm ends at that point, and K data clusters and centroid positions for each data cluster are obtained [27].

The core objective of the K-means algorithm is to minimize the cost function, which is as follows [27]:

$$J = \sum_{j=1}^k \sum_{i=1}^n \text{dist}(x_{ij}, m_j) \quad (1)$$

where $\text{dist}(x_{ij}, m_j)$ represents the distance between the data point x_{ij} and the cluster centroid m_j . The purpose of this calculation is to find a classification method for a set of centroids where the sum of data in each group and the centroid of the group are minimized.

The basic flow of the K-means algorithm is as follows and is illustrated in Fig. 2 below: [17].

- K points are placed in the space represented by the objects being clustered. These points form the initial group centroids (Fig. 2a);
- Each object is assigned to the group that has the nearest centroid (Fig. 2b);
- The positions of the K centroids are recalculated when all objects have been assigned (Fig. 2c);

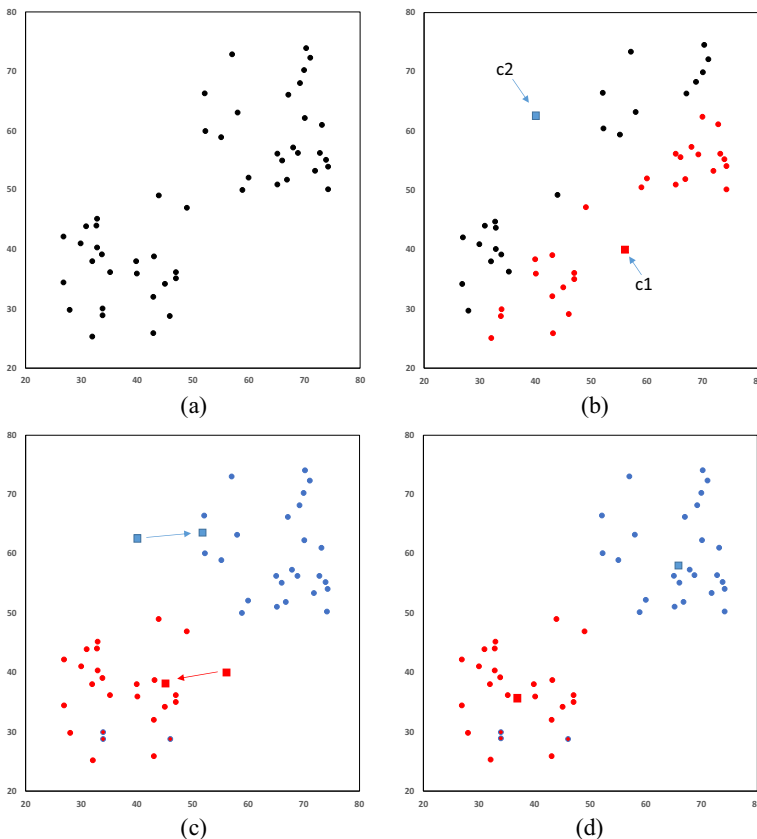


Fig. 2 K-means algorithm implementation process. (a) Initial group centroids (b) Grouping operation (c) Recalculation (d) Repeat (a) and (b) until the centroids no longer move

- d) The degree of change is calculated from the position of the centroid. If the centroid position for each data cluster does not change or the range of variation is less than a certain value, the algorithm ends. Otherwise, repeat Steps (b) and (c) (Fig. 2d).

The K-means algorithm always converges, but it also presents a number of problems [22]. For example, it does not necessarily minimize the optimal clustering of the objective function. Since the K-means algorithm is extremely sensitive to the K value set by the user and the initial centroid position of the K data clusters, this can result in the K-means algorithm objective function eventually converging towards a locally, rather than globally, optimal solution.

3.1.2 K-means algorithm-based image gray level quantization

Image gray level quantization is a process aiming to reduce the number of gray levels within a grayscale picture, that is, using a lower gray level to express an image to reduce the memory overhead of the image. Compared with general gray level quantization, the K-means algorithm can be used to quantize non-uniformly the gray level of the original image. This allows more image details to be preserved with fewer gray levels after image quantization.

Since the grayscale image contains only the luminance features of the image, it is a one-dimensional feature. A grayscale image I_{H*W} with a pixel height of H and a width W is converted into an image $I'_{1*(H*W)}$ with a height of 1 and a width of $H*W$. Sort I' results in I'' . Based on the K value given by the user, the initial centroids of K are randomly set in I'' as $m_{01}, m_{02}, \dots, m_{0K}$. The distance function is defined as follows [3, 27]:

$$\text{Dist}(b, a) = (b-a)^2 \quad (2)$$

To classify the elements in I'' , let $j \in [1, H*W]$, the set of distances from j to centroids is:

$$D_j = \left\{ \text{Dist}(I''(j), m_{02}), \text{Dist}(I''(j), m_{01}), \text{Dist}(I''(j), m_{0K}) \right\} \quad (3)$$

Find $T \in [1, K]$, so that T satisfies:

$$\text{Dist}(I''(j), m_T) = \min(D_j) \quad (4)$$

Then the element $I''(j)$ is the T^{th} class. Thus, I'' is divided into K subsets and recorded as $M_{01}, M_{02}, \dots, M_{0K}$. According to the elements in the subset, the centroid positions are updated. If the initial centroid of the T -class is m_{0T} and contains N_T elements, the centroid obtained after the first update is:

$$m_{1T} = \frac{1}{N_T} \sum_{i=1}^{N_T} M_{0T}(i) \quad (5)$$

After the first update, K centroids are obtained, $m_{11}, m_{12}, \dots, m_{1K}$. For the T^{th} centroid, if there is a certain δ such that $\text{Dist}(m_{0T}, m_{1T}) < \delta$, the update is stopped, otherwise it continues. The K subsets obtained from the first update by eq. (4) are $M_{11}, M_{12}, \dots, M_{1K}$. From eq. (5), the K centroids after the second update are obtained: $m_{21}, m_{22}, \dots, m_{2K}$. With the final update count being n , the final K centroids are $m_{n1}, m_{n2}, \dots, m_{nK}$. The K subsets of I'' are $M_{n1}, M_{n2}, \dots, M_{nK}$, and the element values in each subset are replaced by corresponding centroid values. This effect is mapped into the original grayscale image I_{H*W} . Finally, a grayscale image I'''_{H*W} which is quantized into K gray scales is obtained. The results of the

gray level quantization of the K-means image at a random starting point when $K = 16$, $K = 8$, $K = 4$, and $K = 2$ are shown in Fig. 3 below.

Although the K-means algorithm simplifies the non-uniform gray level quantization, the choice of K value is subjective, which could adversely affect the quantified results. Moreover, the choice of the initial centroid affects the number of updates and convergences of the K-means algorithm. That is, the K value of the K-means algorithm and the choice of initial centroid would eventually affect the storage space of the gray image and the operating efficiency of the algorithm. It is therefore necessary to find a way to reduce the K value and the adverse effect of the initial centroid K-means algorithm in the gray level quantization problem.

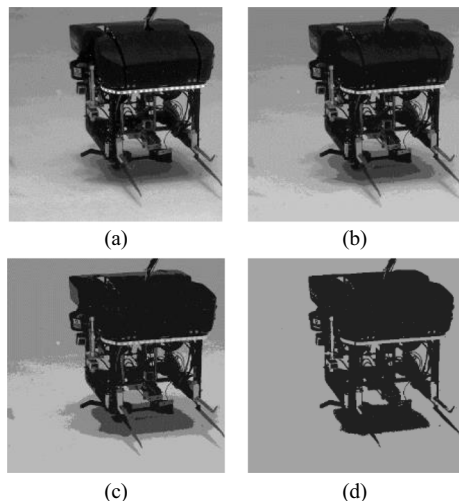
3.1.3 Improved gray level quantization of images based on K-means algorithm

To address the issue of the K value determination and the initial centroid position of grayscale images during the gray level quantization using the K-means algorithm, the present study proposed an improved method of applying gray level quantization of images to reduce the influence of the user's ascribed K value.

K value estimation The K value ultimately reflects the number of gray levels once the gray level of the image has been quantized. By comparing the image histograms before and after quantization, it becomes readily apparent that the K value is related to the histogram of the grayscale image. Since the image histogram reflects the frequency of occurrence of each gray level within the image, the degree of gray level processing in effect uses the most frequently occurring gray level in the original picture to represent the least frequently occurring gray level. Therefore, the pre-quantization image histogram can be used to estimate the number of gray levels requiring quantization, and the estimated number of gray levels is assumed as the K value.

In the present study, the numbers of grayscale pixels of the original grayscale image were counted to obtain a histogram $Hist$. The data in the histogram $Hist$ were blurred to obtain $Hist'$. If $i \in [0, 255]$, and the blur radius is l , then:

Fig. 3 K-means gray level quantization results: (a) $K = 16$; (b) $K = 8$; (c) $K = 4$; (d) $K = 2$



$$Hist'(i) = \sum_{j=i-l}^{2l+1} Hist(i) \tag{6}$$

The number of local histogram maxima obtained after blurring produced the K value. $Diff(i)$ was defined as follows:

$$Diff(i) = Hist'(i + 1) - Hist'(i) \tag{7}$$

If $s \in [0, 255]$, and if a_s was present, then:

$$\begin{cases} Diff(a_s) \cdot Diff(a_s + 1) < 0 \\ Diff(a_s) > 0 \end{cases} \tag{8}$$

In this case, a_s was the maximum value in $Hist'$. The maximum values of $Hist'(i)$ were a_1, a_2, \dots, a_s , where $K = s$.

Figures 4a-c below show the histogram of the input image, the histogram after the mean blur, and the K value estimation respectively. The number of blue peaks indicates the number of quantization, namely where $K = 4$.

Target gray level estimation of K-means The K-means algorithm was used to perform gray level quantization. The mean of each subset was used instead of the subset element values to achieve the final gray level quantization result. The initial centroid value could therefore be estimated by using the peak interval of the maximum value of the original histogram.

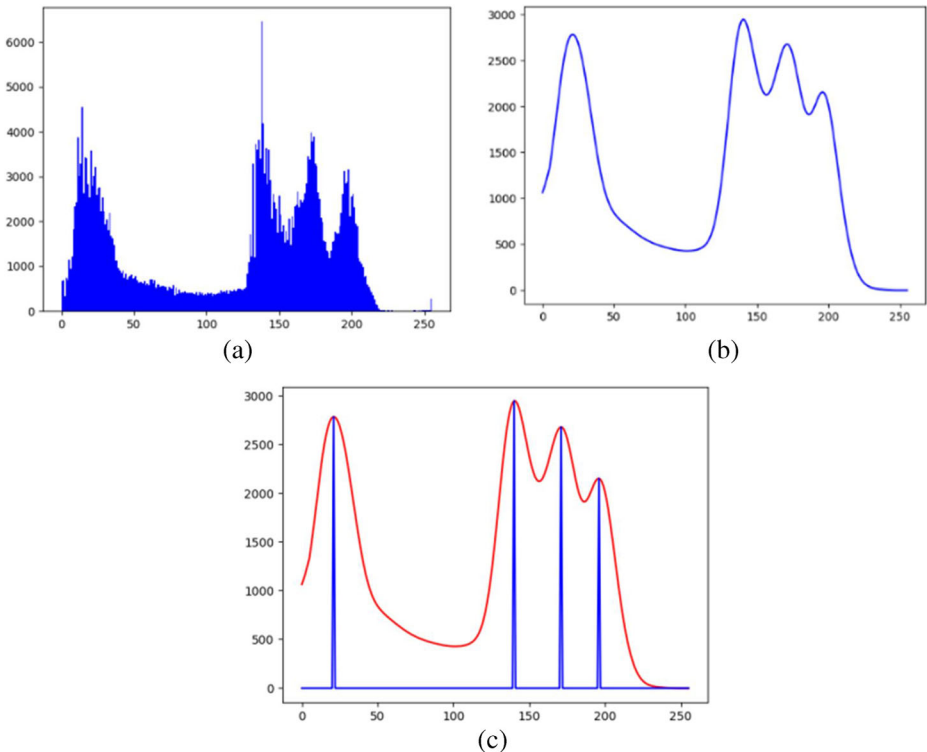


Fig. 4 Illustration of (a) histogram of an image; (b) histogram after the mean blur; and (c) K value estimation

If $t \in (0, 255)$ and if b_t exists:

$$\begin{cases} Diff(b_t) \cdot Diff(b_t + 1) < 0 \\ Diff(b_t) < 0 \end{cases} \tag{9}$$

Then, b_t is the minimum value in $Hist'$. The minimum values of $Hist'$ (i) are b_1, b_2, \dots, b_t . Then, each peak interval is $[0, b_1], [b_1 + 1, b_2], \dots, [b_t + 1, 255]$.

Let $p \in [1, K]$, then the p^{th} peak interval is $[c_p, d_p]$. Then, the p^{th} centroid estimate is:

$$m'_p = \frac{\sum_{i=c_p}^{d_p-c_p+1} Hist(i) * i}{\sum_{i=c_p}^{d_p-c_p+1} Hist(i)} \tag{10}$$

The centroid estimation values of the K centroids were obtained from eq. (10) above, which were denoted as m'_1, m'_2, \dots, m'_K . The centroid estimate was then taken as the initial centroid value for the K -means algorithm.

Figure 5 below shows the result of K value estimation using the result of the input image histogram mean and performing centroid estimation. The number of blue peaks indicates that of quantizations, namely where $K = 4$. Figure 6 below illustrates the estimated quantized image histogram (yellow) and a histogram of the actual quantized image (blue).

Figures 6 (a) (b) below show the gray level quantization results obtained by the random centroid and the proposed method respectively. By way of comparison, the authors' method provided the capacity to converge to the globally, as opposed to the locally, optimal solution, and eventually obtain the ideal quantitative result, as shown in Fig. 7b below.

3.2 Background judgment and segmentation with parameter settings

Underwater image backgrounds present distinctive features. For example, a background can be dark blue, and the area can be smoother. These characteristics enabled the authors to determine whether the quantized area of each gray level pixel was the background image area.

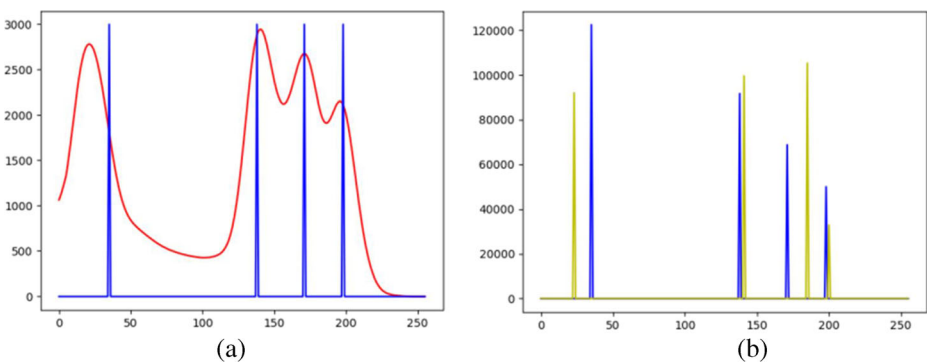


Fig. 5 **a** Centroid estimation; and **b** estimated quantized image histogram (yellow) alongside actual quantized image (blue)

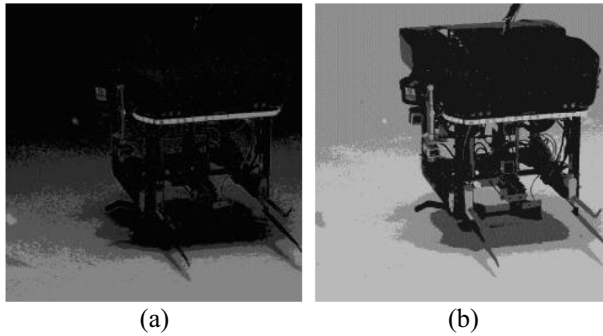


Fig. 6 Image quantization results obtained by (a) the random centroid, and (b) the proposed method

3.2.1 Background decision based on image background color features

The grayscale pixels of the quantized image were separated to form each grayscale mask, $Mask_1, Mask_2, \dots, Mask_n$. The color-adjusted image was converted to the HSV color space, which was defined by chromaticity (H), saturation (S), and brightness (V). The chrominance (H) channel value was $[0, 360]$, which represented the wavelength color change from red to blue light. The standard blue was 240, and the blue field was $[180, 300]$. The pixel chrominance value was used to determine whether it was a background pixel, and the positive membership function of the background pixel was defined:

$$b(x) = e^{-\left(\frac{x-240}{120}\right)^2} \quad (11)$$

where x was the pixel chromaticity value. The chrominance images I_1, I_2, \dots, I_n of the respective quantized gray level regions were obtained by performing an AND operation with each mask and a chroma channel image of the color image within the HSV color space. The background membership degree of the pixels in the image obtained by each mask operation was obtained, and the mean value of the background membership of the K^{th} image was calculated as follows:

$$B(k) = \frac{1}{N} \sum_{i=0}^H \sum_{j=0}^W Mask_k(i, j) b[I_k(i, j)] \quad (12)$$

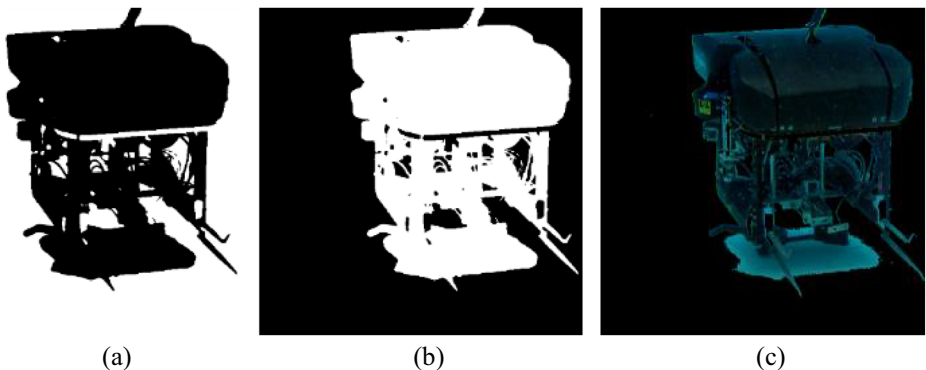


Fig. 7 Illustration of (a) background mask; (b) foreground mask; (c) foreground image

where N was the pixel number of $Mask_k$, H and W were the image height and width respectively. The value of λ was 0.7 if the condition below was met:

$$B(k) > \lambda \quad (13)$$

$Mask_k$ was then the color decision condition for the region to satisfy the background image requirements.

3.2.2 Background decision based on image background frequency characteristics

The degree of image smoothness depends the frequency domain property of the image. Generally, the image of a foreground object is rich in edge information, pixel gray levels change significantly, the background image is quite smooth, and the gray level variation is relatively uniform. In this study, the authors determined whether the region represented a background by analyzing the gray level of each pixel within a given image area, and the gray level of the surrounding pixels.

The average gray variance product (SMD2) function was furthermore used as the basis for evaluating the smoothness of a given image region. The corresponding gray images of each gray level mask, $Mask_1$, $Mask_2$,, $Mask_n$ and the original color image were utilized to perform the AND operation and obtain the gray images I_1 , I_2 ,, I_n for each quantized gray level region. The average grayscale product of each mask and the computed image were:

$$SMD2(k) = \frac{1}{N} \sum_{i=1}^{H-1} \sum_{j=1}^{W-1} Mask_k(i, j) |I_k(i, j) - I_k(i+1, j)| |I_k(i, j) - I_k(i, j+1)| \quad (14)$$

where N was the pixel number of $Mask_k$ and H and W represented the image height and width respectively. If the condition $\sigma = 10$ was met:

$$SMD2(k) < \sigma \quad (15)$$

then, $Mask_k$ was the frequency judgment condition for the region to satisfy the background image requirements.

3.2.3 Background decision based on image background airspace features

In general, pixel location pertains to structural image space information, the background image always surrounds the foreground object, and is at the edge of the image. Based on these characteristics, the position of each gray level mask pixel was deviated from the central pixel point by defining average eccentricity, thereby determining whether the corresponding image area of the mask was the background. As the grayscale masks were in the order $Mask_1$, $Mask_2$,, $Mask_n$, the average eccentricity of the K^{th} mask $Mask_k$ was:

$$S(k) = \frac{1}{N} \sum_{i=0}^H \sum_{j=0}^W Mask_k(i, j) \cdot \sqrt{(i-I)^2 + (j-J)^2} \quad (16)$$

where N was the pixel number for $Mask_k$, H and W indicated the image height and width respectively, and I, J was the image center pixel coordinate. If the condition $\delta = 200$ was met:

$$S(k) > \delta \quad (17)$$

then $Mask_k$ represented the airspace decision condition for the region to satisfy the background image.

3.2.4 Background segmentation

If the conditions (13), (15), and (17) were all satisfied, the mask was described as $Mask_{k1}$, $Mask_{k2}$,, $Mask_{kn}$, and the background mask $Mask_0$ was:

$$Mask_0 = \bigvee_{i=1}^n Mask_{ki} \quad (18)$$

Once the background mask $Mask_0$ was obtained, the foreground mask $Mask'_0$ was:

$$Mask'_0 = I_{ones} - Mask_0 \quad (19)$$

where I_{ones} was the matrix of ones. The respective channels of the color image were compared with the foreground mask to obtain a post-background segmentation image. As an example, Figs. 7(a), (b), and (c) below show the background mask obtained by using Eq. (18), the foreground mask obtained by Eq. (19), and the foreground image obtained by using the foreground mask respectively:

4 Experimental evaluation

4.1 Data description and testing procedure

A total of 100 images including 80 with different types of color board and 20 of various underwater scenes taken by an underwater robot were used to demonstrate the effectiveness of the proposed algorithm. The 100 images were obtained in accordance with the following procedure: (1) placing of color boards in water; (2) mounting of a camera to an underwater robot used to collect images of color boards or underwater scenes at various angles and distances; (3) randomly extract 80 samples from the collected color board images and 20 samples from the collected underwater scene images; (4) apply the developed algorithm to process images and compare results with other methods, as shown in Section 3.2.3 above. Due care was taken to ensure the images were representative and human error was mitigated. Figures 8 and 9 below illustrate typical images of color board and underwater scenes.

4.2 Evaluation metrics

The authors' experiment aimed to demonstrate that their algorithm was effective and inexpensive. The background segmentation effect was determined subjectively and was generally judged on the basis of perceived degree of satisfaction, thus making the resulting judgment tentative. To improve judgment objectivity, the authors used the quantitative index of background segmentation validity below proposed by Ning et al. [28]:

$$G = 100 - \frac{A(M_0 - M_s)}{A(M_s)} \cdot 200 \quad (20)$$

where M_0 was the mask obtained by the method herein described, M_s was the standard mask obtained by the manual stroke method, $A(M_s)$ was the pixel number of the standard mask, and $A(M_0 - M_s)$ was the difference between M_0 and M_s in the pixel number. It should be noted that if the algorithm generated a template consistent with the standard template, the validity evaluation function (20) result would have been 100. Had the difference between algorithm

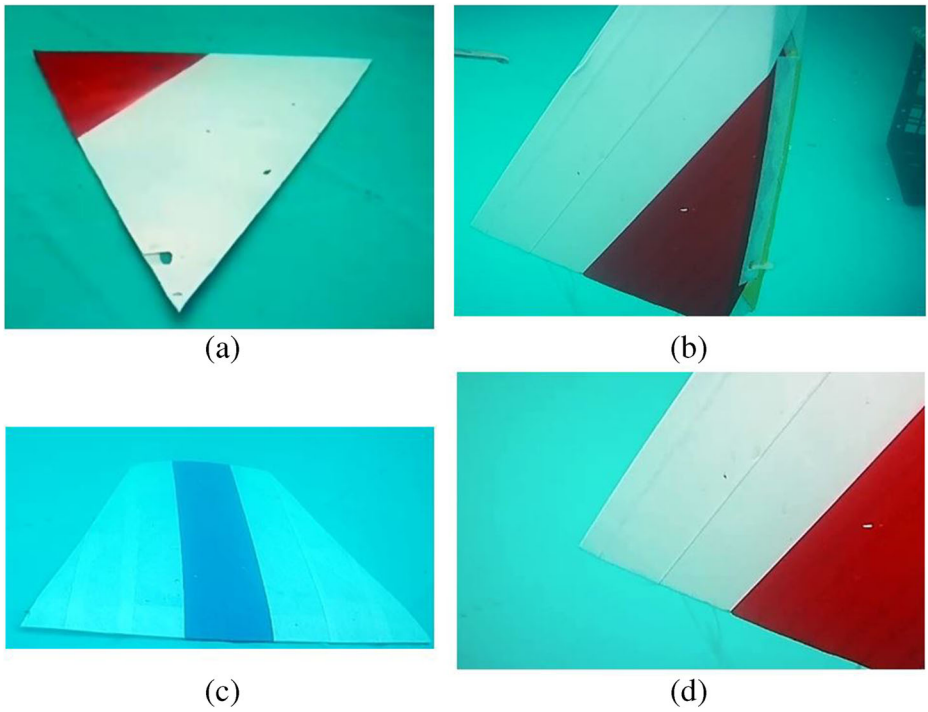


Fig. 8 Typical underwater images of color boards (a, b, c, d) taken by an underwater robot in this study

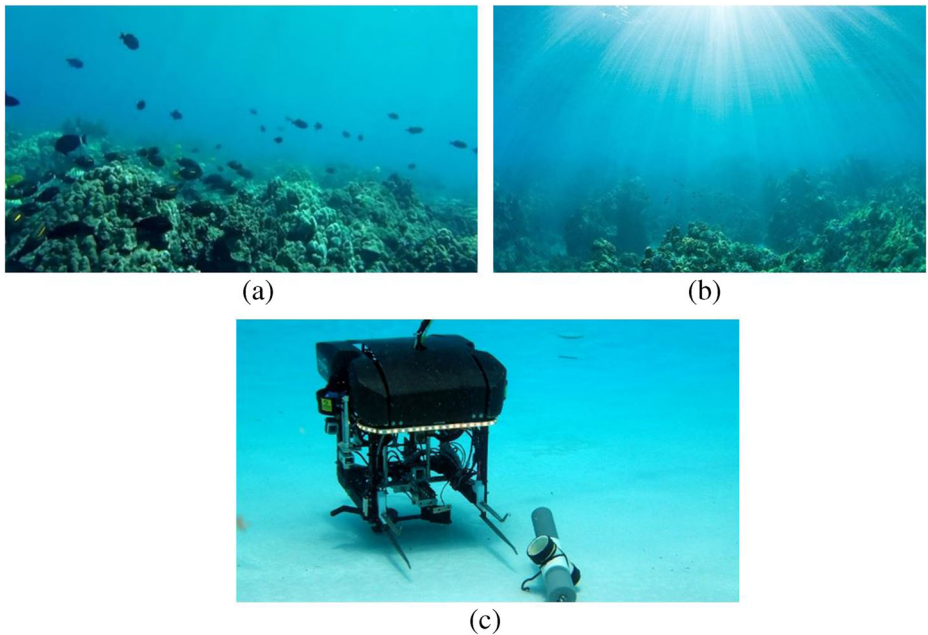


Fig. 9 Typical underwater images of scenes (a, b, e) taken by an underwater robot in this study

template and standard template been greater than 20%, the result of Eq. (20) would have been below 60.

The present study also used a timeliness evaluation function proposed by Cao et al. [8] to evaluate the time cost of the algorithm:

$$T = \frac{t}{s} \quad (21)$$

where t was the total algorithm running time (unit: ms), s was the total number of image pixels, and T was the average pixel processing time of the algorithm.

The algorithms and related experiments described in this paper were implemented on the Win10 computer operating system using the Python 2.7 language and OpenCV open source processing library. The computer hardware was Intel^(R) Core^(TM) i7-4700HQ CPU@2.40 GHz, 2.4 GHz, with 8.00 GB of memory.

4.3 Comparative approaches

The authors' proposed algorithm was compared to the other three state-of-the-art background segmentation approaches widely applied for comparison in different studies. These included:

- The conventional K-means algorithm.
- The improved Otsu algorithm based on simulated annealing genetic algorithm [33]. This improved Otsu algorithm had the capacity to address the drawbacks of slow speed and large memory requirements.
- The Canny operator edge extraction method [34]. This algorithm was capable of excluding fake edges with anomalies and noise.

5 Results and discussion

Figure 10 below illustrates the whole image background segmentation process from initial image as an input to final foreground object segmented by the improved K-means algorithm proposed in this research. An image comprising interior boundaries, low color cast, blurred edges, and poor contrast between object and background was analyzed and presented to demonstrate the effectiveness of the authors' proposed new algorithm. The representative results showed that the foreground object had distinct exterior edges and interior boundaries despite small strips having been found within the interior boundary region. The finite contrast histogram equalization algorithm facilitated the successful identification of boundaries [37]. The above result indicates the improved K-means gray level quantization background removal

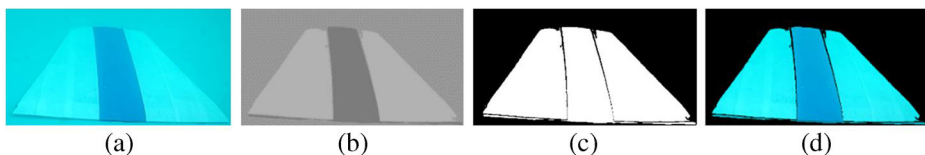


Fig. 10 Image background segmentation process: (a) initial image; (b) image obtained by the K-means algorithm; (c) foreground mask obtained by the improved K-means algorithm; and (d) foreground object obtained by the improved K-means algorithm

algorithm, can effectively remove the underwater image background, of which the color is similar to the main part of the object, to obtain the foreground object image.

The comparative results regarding the foreground masks obtained by the improved K-means algorithm and other methods are presented in Fig. 11 below, where the capacity of the improved K-means algorithm to extract the foreground mask is demonstrated (Fig. 11d). While the improved Otsu algorithm (Fig. 11a), the Canny operator edge extraction (Fig. 11b), and the conventional K-means algorithm (Fig. 11c) failed to capture the complete object.

Figure 12 below reveals the foreground objects extracted via different methods. In line with the above results, the proposed method of this study – namely the improved K-means algorithm – was found effectively to remove the background of underwater images (Fig. 12d) by applying a non-uniform gray level quantization method that could distinguish a foreground object from its background. In contrast, segmentation results obtained from the improved Otsu algorithm (Fig. 12a), the Canny operator edge extraction (Fig. 12b), and the conventional K-means algorithm (Fig. 12c) proved incomplete and unattractive.

Table 1 below illustrates the effectiveness and timeliness (average pixel processing time) of each algorithm, calculated by using Eqs. (20) and (21), including the improved Otsu algorithm, the Canny operator edge extraction method, the conventional K-means algorithm, and the improved K-means algorithm. Compared to other algorithms, the average time cost of our new proposed algorithm was significantly higher. This was also reflected in Fig. 13, where the average pixel processing time of the improved K-means algorithm was compared with the other three algorithms. Although the average time cost of the improved K-means algorithm proved greater than that of the other methods, it generally remained below 1.0 ms and was thus acceptable. It should be noted that the relatively slow speed of the K-means clustering algorithm had previously generally been recognized by other researchers [22]. However, the proposed K-means algorithm proved significantly faster than the manual segmentation method when processing large quantities of underwater images.

The background segmentation validity results in Fig. 14 below reveal slight fluctuations in the background segmentation validity values of the authors' improved K-means algorithm,

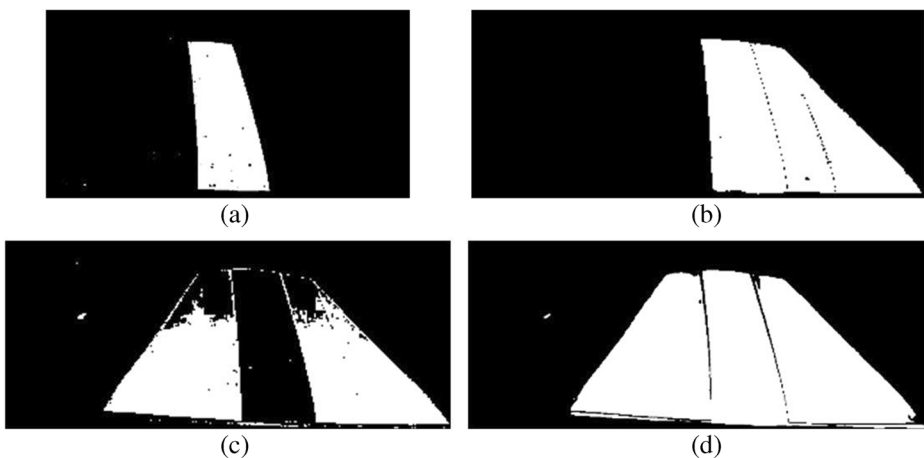


Fig. 11 Comparison of foreground masks obtained via different segmentation methods: (a) improved Otsu algorithm; (b) Canny operator edge extraction; (c) conventional K-means algorithm; and (d) improved K-means algorithm

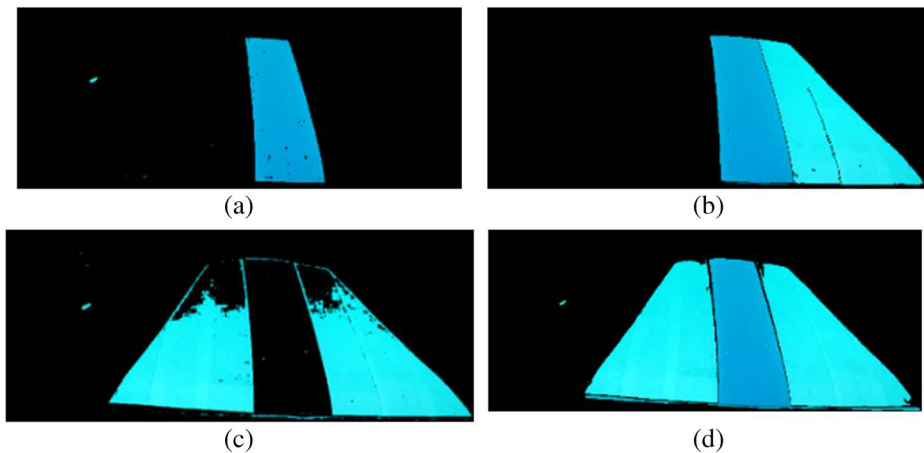


Fig. 12 Comparison between foreground objects obtained by different segmentation methods: (a) improved Otsu algorithm; (b) Canny operator edge extraction; (c) conventional K-means algorithm; and (d) improved K-means algorithm

with an average value of around 90 and standard deviation of 4.9. The average validity value and the standard deviation value are respectively significantly higher and lower than those of the other three algorithms. The above results demonstrate that the improved K-means gray level quantization background segmentation algorithm is superior to the other conventional image background segmentation algorithms examined in the present study.

Underwater images are subject to non-uniform brightness, poor contrast, and diminished colors, as shown in the typical underwater images (Figs. 8 and 9 above). These factors can result in the failure of current background segmentation methods based on histogram threshold and edge detection techniques, which generally require high grey level contrast of object boundaries. While the K-means algorithm was shown to improve the performance of the background segmentation algorithms based on histogram threshold segmentation and that edge detection respectively, the impact of initial centroid on segmentation results produced local, rather than global, optimization. The authors' improved K-means algorithm provided the capability of specifically processing underwater images, the gray levels of which pixels were similar to those of the foreground image pixels. The issue of improper K value determination was thus resolved, and the impact of initial centroid position of grayscale image during the gray level quantization of the conventional K-means algorithm was reduced.

Underwater images furthermore commonly incur a low color cast, low contrast, and blurred edges. The segmentation results in this study were promising as interior boundaries of images could be clearly identified while whole foreground objects could readily be segmented (Fig. 12

Table 1 Algorithm effectiveness and timeliness

Segmentation method	Effectiveness	Average pixel processing time / ms
Improved Otsu algorithm	-52.39	0.016
Canny operator edge extraction method	19.13	0.084
Conventional K-means algorithm	19.58	0.166
Improved K-means algorithm	90.34	0.568

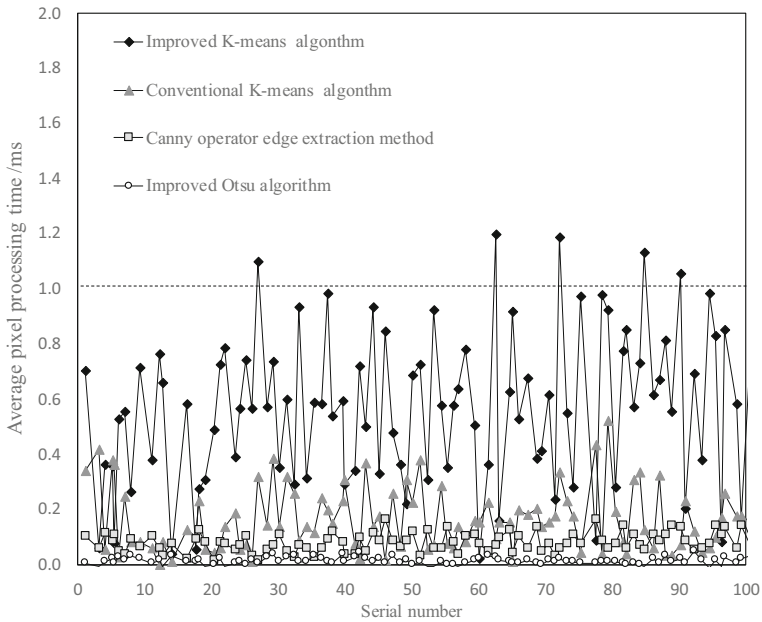


Fig. 13 Average pixel processing timeline graph based on each background segmentation algorithm

above). According to Ning et al. [28], the quality of background segmentation validity was significantly higher than that of other image background segmentation algorithms – namely the improved Otsu algorithm, Canny operator edge extraction, and conventional K-means

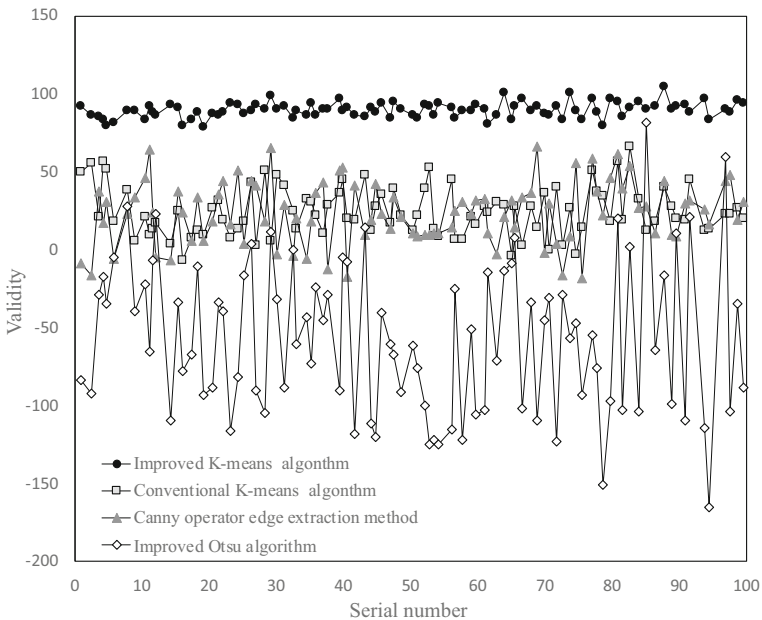


Fig. 14 Validity line graph of image segmentation based on each background segmentation algorithm

algorithm – suggesting potential for the authors’ underwater background segmentation algorithm to be applied in real underwater environments. Further research remains called for in order to further optimize K value estimation algorithm and centroid estimation algorithm to enhance the image processing speed.

6 Conclusion

In light of the failure of existing background segmentation models to achieve accurate object segmentation on underwater images due to the impact of light absorption and dispersion on the characteristics of underwater images, an improved K-means algorithm was developed in the present study to enhance underwater image background segmentation. New algorithms were elaborated to tackle the issue of improper K value determination and reduce the impact of initial centroid position of images during the gray level quantization of the conventional K-means algorithm. A dataset of 100 underwater images taken by an underwater robot was used to test the proposed algorithm in terms of its background segmentation validity and time cost. The results were compared to other state-of-the-art algorithms, including the conventional K-means algorithm, the improved Otsu algorithm, and the Canny operator edge extraction method. The following conclusions were drawn:

- The improved K-means underwater background segmentation algorithm produced a significantly higher effectiveness value of 90.34 compared to the other three segmentation methods, the effectiveness values of which were below 20.00. The image analysis results also demonstrated that the new model could successfully segment the background of underwater images, which were generally prone to low color cast and contrast, and blurred edges. In contrast, the conventional K-means algorithm, improved Otsu algorithm, and Canny operator edge extraction method often failed to differentiate and extract foreground objects.
- The average pixel processing time of the improved K-means algorithm was measured at around 0.568 ms, i.e. under 1.0 ms. Although significantly greater than that of other algorithms, this average pixel processing time remained considerably below that of the manual segmentation method, and was thus acceptable. Future research could be aimed at optimizing the K value estimation and centroid estimation algorithms respectively in order to enhance the image processing speed.
- The K-means algorithm developed herein demonstrably overcomes the challenges of K value determination and centroid position during gray level quantization. It thus offers the potential to be applied to detect underwater image objects in challenging underwater environments for a wide range of security applications.

Acknowledgments The authors appreciate the funding supported by the Nanjing Industry-University-Research Cooperation Funding Project (No. 221722072).

Funding Open Access funding enabled and organized by Projekt DEAL. This research was supported by the Nanjing Industry-University-Research Cooperation Funding Project (No. 221722072).

Code availability The code required to reproduce these findings cannot be shared at this time as it also forms part of an ongoing study.

Declarations

Conflict of interest The authors declare that there is no conflict of interest regarding the publication of this paper.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Abutaleb AS (1989) Automatic thresholding of gray-level pictures using two-dimensional entropies. *Pattern Recogn* 47(01):22–32. [https://doi.org/10.1016/0734-189X\(89\)90051-0](https://doi.org/10.1016/0734-189X(89)90051-0)
2. Ahmad A, Dey L (2007) A k-mean clustering algorithm for mixed numeric and categorical data. *Data Knowl Eng* 63(2007):503–527. <https://doi.org/10.1016/j.datak.2007.03.016>
3. Al-jabery K, Obafemi-Ajayi T, Olbricht GR, Wunsch II DC (2019) 3-clustering algorithms. *Computational Learning Approaches to Data Analytics in Biomedical Applications Academic Press* <https://doi.org/10.1016/C2016-0-04633-8>
4. Arvis V, Debain C, Berducat M, Benassi A (2004) Generalization of the cooccurrence matrix for colour images: application to colour texture classification. *Image Anal Stereol* 23(1):63–72
5. Bazeille S, Quidu I, Jaulin L, Malkasse J (Oct 2006) Automatic underwater image pre-processing. *CMM'06, Brest, France*. hal-00504893
6. Bo S, Yan M, Sun G et al (2007) Research on crack detection image processing algorithm for asphalt pavement surface. *J Microcomput infor* 23(15):280–282
7. Cao Y, Zhao J, Yan J (2009) Image segmentation method based on custom color space model. *Comput Sci* 36(02):265–267
8. Cao L, Dai Q, Pan Q (2012) Improved background removal based on canny algorithm and threshold segmentation. *Comput Eng App* 48(01):208–211
9. Chan MT, Scarafoni DJ, Bockman AC (2019) U.S. patent application no. 15/864, 912.
10. Chen C, Wang J, Zou L et al (2012) Underwater dam image crack segmentation based on mathematical morphology. *App Mech Mater* 2(03):1315–1319. <https://doi.org/10.4028/www.scientific.net/amm.220-223.1315>
11. Chen Y, Xu W, Zuo J, Yang K (2018) The fire recognition algorithm using dynamic feature fusion and IV-SVM classifier. *Cluster Comput* 22:7665–7675. <https://doi.org/10.1007/s10586-018-2368-8>
12. Chen Y, Xiong J, Xu W, Zuo J (2018) A novel online incremental and decremental learning algorithm based on variable support vector machine. *Cluster Comput* 22:7435–7445. <https://doi.org/10.1007/s10586-018-1772-4>
13. Chen Y, Wang J, Liu S, Chen X, Xiong J, Xie J, Yang K (2019) Multiscale fast correlation filtering tracking algorithm based on a feature fusion model. *Concurrency Computat Pract Exper* 2019:e5533. <https://doi.org/10.1002/cpe.5533>
14. Chen Y, Wang J, Xia R, Zhang Q, Cao Z, Yang K (2019) The visual object tracking algorithm research based on adaptive combination kernel. *J Ambient Intell Human Comput* 10:4855–4867. <https://doi.org/10.1007/s12652-018-01171-4>
15. Chen Y, Tao J, Liu L, Xiong J, Xia R, Xie J, Zhang Q, Yang K (2020) Research of improving semantic image segmentation based on a feature fusion model. *J Ambient Intell Human Comput*. <https://doi.org/10.1007/s12652-020-02066-z>
16. Chen Y, Tao J, Zhang Q, Yang K, Chen X, Xiong J, Xia R, Xie J (2020) Saliency detection via the improved hierarchical principal component analysis method. *Wirel Commun Mob Comput* 2020:8822777–8822712. <https://doi.org/10.1155/2020/8822777>

17. Dehariya VK, Shrivastava SK, Jain RC (2010) Clustering of image data set using K-means and fuzzy K-means algorithms. *International Conference on Computational Intelligence and Communication Networks* 2010:386–391. <https://doi.org/10.1109/CICN.2010.80>
18. Du H, Chen X, Xi J (2019) An improved background segmentation algorithm for fringe projection profilometry based on Otsu method. *Opt Commun* 453:124206. <https://doi.org/10.1016/j.optcom.2019.06.044>
19. Fielding S, Copley JT, Mills RA (2019) Exploring our oceans: using the global classroom to develop ocean literacy. *Front Mar Sci* 6:340. <https://doi.org/10.3389/fmars.2019.00340>
20. Gonzalez R (2007) *Digital image processing*, 3rd edn Prentice Hall
21. Hu M, Li M, Wang R (2010) Application of an improved Otsu algorithm in image segmentation. *J Electron Meas Instrum* 24(05):443–449. <https://doi.org/10.3724/SP.J.1187.2010.00443>
22. Jain S, Laxmi V (2018) Color image segmentation techniques: a survey. In: Nath V. (eds) *Proceedings of the International Conference on Microelectronics, Computing & Communication Systems*. Lecture Notes in Electrical Engineering, vol 453. Springer, Singapore. https://doi.org/10.1007/978-981-10-5565-2_17
23. Jiang Y, Li F (2010) Multi-threshold method of image segmentation based on chaotic particle swarm optimization algorithm. *Comput Eng App* 46(10):175–176 <http://cea.ceaj.org/EN/10.3778/j.issn.1002-8331.2010.10.055>
24. Kittler J, Illingworth J (1986) Minimum error thresholding. *Pattern Recogn* 19(1):41–47. [https://doi.org/10.1016/0031-3203\(86\)90030-0](https://doi.org/10.1016/0031-3203(86)90030-0)
25. Li Z, Zhang D, Xu Y, Liu C (2011) Modified local entropy-based transition region extraction and thresholding. *Appl Soft Comput* 11:5630–5638. <https://doi.org/10.1016/j.asoc.2011.04.001>
26. Luo Y, Qin J, Xiang X, Tan Y, Liu Q, Xiang L (2019) Coverless real-time image information hiding based on image block matching and dense convolutional network. *J Real-Time Image Proc* 17:125–135. <https://doi.org/10.1007/s11554-019-00917-3>
27. MacKay DJC (2003) *Information theory, inference and learning algorithms*. Cambridge University Press, Cambridge
28. Ning X, Zhang S, Tan L (2007) Background removal based on canny and thresholds complex segmentation technology. *Chinese J Medi Phy* (05):326–328+355
29. Peng C, Zhang XY, Yu G, Luo JM, Sun J (2017) Large kernel matters-improve semantic segmentation by global convolutional net-work. In: *Proceedings of IEEE conference on computer vision and pattern recognition*, pp 1743–1751. <https://doi.org/10.1109/CVPR.2017.189>
30. Priyadharsini R, Sharmila S (2019) Object detection in underwater acoustic images using edge based segmentation method. *Procedia Comput Sci* 165:759–765. <https://doi.org/10.1016/j.procs.2020.01.015>
31. Rugna JD, Gael C, Hubert K (2011) About segmentation step in content-based image retrieval systems, in *Proceedings of World Congress on Engineering and Computer Science*, vol. 1, 19–21, USA
32. Shelhamer E, Long J, Darrell T (2017) Fully convolutional networks for semantic segmentation. *IEEE Trans Pattern Anal Mach Intell* 39 (4):640–651. <https://doi.org/10.1109/TPAMI.2016.2572683>
33. Sun F, Wang H, Fan J (2009) 2D Otsu segmentation algorithm based on simulated annealing genetic algorithm for iced-cable image. *International Forum on Information Technology and Applications* 2:600–602
34. Tan X, Zhang D, Ma G (2018) Edge detection of potential field data based on image processing methods. *Glob Geo* 21(02):134–142
35. Wu Z, Bu J, Chen C (2002) Dynamic programming cased video segmentation. *J Comput-Aided Des Comput Graphics* 08:743–746+749
36. Xu X, Cai Y, Liu C et al (2008) Color cast detection and color correction methods based on image analysis. *Meas Control Tech* (05): 10–12+21.
37. Yang W, Xu Y, Qiao X et al (2016) Method for image intensification of underwater sea cucumber based on contrast-limited adaptive histogram equalization. *Trans Chinese Soc Agri Eng* 32(06):197–203
38. Yang MK, Yu K, Zhang C, Li ZW, Yang KY (2018) DenseASPP for semantic segmentation in street scenes. *Proceedings of IEEE conference on computer vision and pattern recognition*, In, pp 3684–3692. <https://doi.org/10.1109/CVPR.2018.00388>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Wei Chen received his MSc degree from Fudan University in 2008. Currently, he is an associate professor and Master students' Supervisor of Industrial Center, Nanjing Institute of Technology and Head of the Ministry of Intelligence, Shenzhen Kuang-Chi Space Technology CO.LTD. His research fields focus on intelligent robots and image processing.



Cenyu He is studying for a master's degree. Now he is a student of mechanical engineering in Nanjing Institute of Technology. His main research direction is embedded development.



Chunlin Ji Ph.D in Duke University in 2009, Master's Degree in University of Cambridge in 2006. He is the vice-president in Shenzhen Kuang-Chi Institute of Advanced Technology. His research fields focus on Bayesian statistics, artificial intelligence, computer experiment design, meta-material design.



Meiyang Zhang Ph.D in Mechanical Engineering (Robotics), Laval University, Canada, in 2015. Now she is working in Shenzhen Kuang-Chi Space Technology CO.LTD. Her research fields mainly focus on detection, tracking and recognition in computer vision, artificial intelligence, analysis and implementation of behavior patterns, programming the robot motion and stability analysis.



Siyu Chen is now a postgraduate student at Department of Electrical and Computer Engineering, Technical University of Munich, Germany. Her research fields focus on intelligent robots and image processing.

Affiliations

Wei Chen^{1,2} · **Cenyu He**¹ · **Chunlin Ji**³ · **Meiying Zhang**² · **Siyu Chen**⁴

Wei Chen
chenwei@njit.edu.cn

Cenyu He
851758060@qq.com

Chunlin Ji
chunlin.ji@kuang-chi.com

Meiying Zhang
meiying.zhang@kuang-chi.com

¹ Industrial Center, Nanjing Institute of Technology, College of Innovation and Entrepreneurship, Nanjing 211167, China

² Shenzhen Kuang-Chi Space Technology Co. Ltd, Shenzhen 518000, China

³ Shenzhen Kuang-Chi Institute of Advanced Technology, Shenzhen 518000, China

⁴ Department of Electrical and Computer Engineering, Technical University of Munich, D-80333 Munich, Germany