Automatic gas leak detection system

Shuai Wang, Bin Gong *, Yang Liu, Wankun Li
School of Qingdao University of Science and Technology, Qingdao, China
*gloriabinbin@163.com

Abstract. With the development of submarine oil and gas resources, people pay more and more attention to the safety of pipeline transportation. Once an oil and gas leak occurs in the submarine transportation pipeline, it will not only cause direct economic loss but also damage the marine ecology. Therefore, it is particularly important for underwater gas leak detection research. Due to the difficult underwater detection operation and the high labor cost, an embedded module with automatic gas leak detection function is designed and completed in this paper. This paper analyzes the imaging characteristics of multi-beam sounding sonar images, and designs a set of underwater gas leak detection algorithms. The algorithm mainly includes: median filtering, bottom detection, top-hat processing, maximum entropy image segmentation, connected area labeling, nearest neighbor interpolation based on backward mapping, linear upward trend estimation based on Hough transform, and testing of 120 images, With an accuracy rate of more than 98%.

1. Introduction
With the rapid development of industry and economy in the world, the consumption of oil and gas resources has increased dramatically. With the massive exploitation of offshore oil and gas resources, the transportation of oil and gas has become a top priority. Pipeline transmission is widely used because of its economic efficiency and little impact on the environment [2]. However, most of the gas transmission pipelines are laid on the seafloor with poor environmental conditions, which will not only be impacted by tidal currents, but also be more vulnerable to internal oil and gas corrosion [3]. Moreover, oil pipelines often need to be laid for a long distance and have a long service life. Regular inspection and repair can effectively prevent more serious oil and gas leakage. However, due to its complex laying environment, the detection of oil and gas leakage has become very difficult, so the research on the detection of oil and gas leakage of submarine pipeline has become increasingly important. In recent years, although the research on gas leakage detection is increasing at home and abroad, the problem of gas leakage detection in the marine environment is still a problem. In the detection of oil and gas leakage on land, the most commonly used method is in-line detection. Xi'an University of petroleum once used the principle of water hammer wave to detect the change of pressure in the oil pipeline to determine whether there is leakage [5]. However, in-line detection generally needs to block the pipeline. Because of the high risk of subsea pipeline blocking, and will cost a lot of money, in-line detection is not suitable.

It is used to detect the gas leakage on the sea floor. At present, the gas leakage detection of subsea oil and gas pipeline is more by the method of out of pipe detection. The commonly used out of pipe detection methods are mainly divided into manual and non manual methods. The manual detection method can only detect a specific area. The main way of detection is to find the leakage location in the
designated area through the detection equipment carried by the diving personnel. Generally, the equipment carried is the underwater camera, which is detected by the underwater optical image. However, due to the close underwater shooting distance and the safety problems of diving personnel, long-distance detection cannot be carried out, which is only suitable for short-term detection, but not for long-distance transportation pipelines. Therefore, this paper designs a set of automatic monitoring system to better solve the shortcomings of long-distance transportation and high cost. The first paragraph after a heading is not indented (Bodytext style).

2. Design principle of automatic detection system

2.1. Sonar Image Data Analysis

When the acoustic signal propagates in the water, it will produce attenuation scattering, and it will also be affected by the noise in the water. Therefore, the image data from the multi beam sounding sonar often has the characteristics of low SNR and low contrast. Based on the imaging characteristics of multibeam sounding sonar, this paper first designs a set of image preprocessing process to improve the image quality, then extracts the suspected leakage area through segmentation, connected area marking and other operations, and finally judges whether there is gas leakage by estimating the linear rising trend of the area.

The multi-beam sonar image data used in this paper is derived from multi-beam sounding sonar. It uses the snapshot method as the imaging method, that is, imaging the complete echo signal recorded by sonar\[6\]. The complete echo signal is capable of imaging both the seafloor area and the water body. The sonar image is generally a grayscale image with only one grayscale channel. A multi-beam sonar transmits a set of fan beams to the ocean floor, and then obtains the position data of each sampling point and the corresponding directional scattering intensity, which can be represented as a gray value on the image. When using multi-beam sounding sonar to detect gas, the backscattering intensity when the beam hits the gas is strong, and it will appear brighter in the image than the surrounding environment. Therefore, multi-beam sounding sonar can be used for gas in water Imaging detection.

The sonar image data is generally presented in a rectangular manner. As shown in Figure 2.1, highlighted areas when gas leaks\[7\]. In the image, the beam number is used as the abscissa, and each beam sampling time is used as the ordinate. The image resolution used in this paper is 255 * 1500, that is, there are 255 beams, each beam has 1500 sampling points, and the sampling frequency is 48KHZ. The scene for data collection is a pool. Because the actual imaging of the multi-beam sounding sonar should be a fan shape, the bottom of the pool will now appear curved after being displayed in a rectangular manner. It needs to be restored to a fan-shaped display in the subsequent processing, as shown in Figure 2.2. When the acoustic signal is transmitted in water, attenuation and scattering occur, in addition to being affected by sonar self-noise and bubbles and reverberation noise. From Figure 2.1, we can see that the sonar image is accompanied by severe noise interference. The noise generated in the image is close to the target area, resulting in a very low contrast in the image.
2.2. Design Ideas of Gas Leak Detection Algorithm

Sonar images are susceptible to the complex underwater environment, and due to the attenuation and scattering of acoustic signals as they propagate in the water, the sonar images generally obtained will have very serious noise interference and have a low signal-to-noise ratio. So first we need to filter and denoise the image to reduce noise interference. The data under the pool bottom in the image is useless. In order to reduce the processing of image data and the interference of the pool bottom, the bottom area needs to be detected and then the bottom area is removed. Because the sonar image is imaged with echo signals, the image quality is often poor and has low contrast. In order to better segment the foreground and background information of the image and extract the target area of interest, the contrast needs to be enhanced by image enhancement. Next, an image segmentation method can be used to segment the image into two parts: foreground information and background information. After the segmentation is completed, it is necessary to extract the suspected gas leakage area from the foreground information and estimate the rising shape, so as to determine whether it is a gas leakage area. The overall design flow of the algorithm is shown in Figure 3. The rest of this chapter will specifically introduce the algorithm selected for each process, and the effect comparison between different algorithms.
3. Specific Design Process

3.1. Image filtering
Multi-beam sounding sonars are imaged by echo signals, but there are many noise sources in the complex underwater environment. E.g: Reverberation noise, environmental noise, mechanical noise, etc., often cause sonar images to be accompanied by severe noise interference, which has a lower signal-to-noise ratio [9]. Noise will appear as bright spots in the sonar image, and its gray value is close to the target gray value, making the image blurred and even affecting the target features. If noise is not suppressed, it is not conducive to subsequent processes such as bottom detection, image segmentation, and target region extraction. Therefore, it is necessary to suppress image noise, improve the signal-to-noise ratio of the image, and ensure that subsequent processing proceeds smoothly. Noise in images generally has three characteristics [10]: Randomness of noise distribution in the image, correlation between noise and image, and superposition of noise itself. The suppression of noise in the image is generally filtered. The filtering of the image is mainly divided into spatial filtering and frequency domain filtering. Because the final algorithm needs to be implemented in the C language on an embedded platform, the implementation process of frequency domain filtering is more complicated and depends on more library functions, which are not suitable for embedded implementation. Therefore, we mainly discuss several aspects in spatial filtering. Different algorithms.

3.1.1. Mean filtering. Spatial domain filtering refers to performing operations on the original image and processing pixel gray values based on the characteristics of surrounding gray values. It can be divided into linear filtering and nonlinear filtering. Mean filtering is a classic linear filtering algorithm [11]. The idea of the algorithm is to replace the value of itself with the gray value of several points near the point.

Mean filtering must first give an operation template, which contains the target gray value and neighboring gray values that need to be operated. Then replace the target gray value with the average of all gray values in the template. Table 1 shows an example of a $3 \times 3$ template.

**Table 1.** Mean Filtering Template 3 * 3

| One  | Four  | Seven |
|------|-------|-------|
| Two  |       | Eight |
| Three| Six   | Nine  |

Set the target gray value to $g(x, y)$, select a template, and then find the average value of all gray levels in the template. Use this average value as the gray value of that point in the image after the operation, that is,

$$g(x, y) = \frac{1}{m} \sum_{f(x, y) \in \text{template}} f(x, y)$$  \hspace{1cm} (1)

In the expressions is the set of template elements; $m$ is the number of pixels in the template; and $g(x, y)$ is the gray value of the elements in the template.
3.1.2. Median Filter. Median filtering is a very typical non-linear filtering method. It is a filter based on statistical ordering. It is based on the ordering of all gray values in the template and replaces the original pixel with the median in the ordered queue. Because it can not only effectively remove impulse noise, but also protect sharp edges of the image, this filtering method is widely used.

The method of median filtering is to select a certain sliding template, sort all the gray values in the template monotonically, and output the sorted median value as the gray value of the point. Set the target gray value to \( g(x, y) \):

\[
g(x, y) = \text{med}\{f(x-k, y-l), (k, l \in s)\}
\]

Where\( f(x, y) \) is the value of the original image, \( s \) is the template. Templates are generally chosen to be odd, such as \( 3 \times 3 \times 5 \) Waiting template. E.g. pick one \( 3 \times 3 \) Template, Gray values in the template \{20,30,30,25,30,30,35,100\}. Sort it \{20,25,30,30,30,30,30,35,100\}, The median is 30, Replace the target gray value with 30. This can make points with different gray levels closer to their neighboring points, which is more conducive to removing independent noise.

3.1.3. Analysis and comparison of two denoising algorithms. Both algorithms can suppress the noise to some extent. The processing results are shown in Figure 3.2. Figure (a) is the original image, Figure (b) is the image after mean filtering, Figure (c) is the image after median filtering.

![Figure 4. Filtering effect](image)

By observing the effect diagrams of the two filters, it can be clearly seen that the suppression effect of the median filter on the noise is better than the mean filter. In addition to subjective perception, the objective criterion for evaluating the filtering effect of an algorithm in an image is to judge the degree of distortion between the noise-reduced image and the original image. Commonly used judgment standards include mean square error (MSE) and peak signal-to-noise ratio (PSNR) [12]. The mean square error is mainly used to judge the degree of deviation between the image after noise reduction and the original image. The larger the mean square error is, the larger the difference between the noise-reduced image and the original image is, which may cause the image during the filtering process. The information except noise has changed, and the filtering effect is poor. Conversely, the smaller the mean square error, the better the filtering effect.

\[
MSE = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} [f(m,n) - f'(m,n)]^2
\]
M is the image width, N is the height of the image, \( f(m, n) \) is the original image, and \( f'(m, n) \) is the image after denoising.

The peak signal-to-noise ratio (PSNR) of an image is an objective criterion for evaluating the results of image filtering processing, which is mainly used to measure the quality of the image after filtering processing. The larger the PSNR, the smaller the distortion of the processed image and the better the quality.

\[
\text{PSNR} = 10 \log_{10} \frac{\text{MAX}(x, y)^2}{\text{MSE}} \quad (4)
\]

\( \text{MAX}(x, y) \) is the maximum gray value in the image, and MSE is the mean square error. The mean square error and peak signal noise of the two algorithms are shown in Table 2.

### Table 2. Comparison of filtering algorithms

| Filtering algorithm | Mean filtering | Median filtering |
|---------------------|----------------|------------------|
| MSE                 | 1.6238         | 1.3523           |
| PSNR                | 40.2436        | 42.3621          |

It can be seen from the table above that the mean square error and peak signal-to-noise ratio of the median filter are better than the mean filter. The distortion after filtering is small, and the original image information can be kept as much as possible while effectively suppressing noise. To sum up, the last filtering algorithm selected in this paper is median filtering.

### 3.2. Bottom area detection

Pipelines are usually laid on the ocean floor. When multi-beam sounding sonar imaging, some of the data below the ocean floor are displayed on the image. The curved part in Figure 2.1 is the bottom of the pool. There is still some imaging data below the bottom of the pool in the image. Because the target gas will only exist on the bottom of the pool, the analysis of the data below the bottom of the pool will become meaningless. In order to reduce the subsequent data processing and eliminate the interference from the bottom area, the bottom of the pool needs to be detected and removed. This can not only greatly reduce the amount of calculation but also remove the interference from the wall of the pool.

In multi-beam sounding sonar midsole detection, conventional beam forming methods and amplitude-phase combined detection methods are mainly used [13]. Because the image data only contains amplitude and beam angle information, the bottom detection algorithm based on amplitude is used here. Fritz Albregtsen proposed in the article on multi-beam seafloor detection. Because the intensity of sound scattering from the seafloor is the strongest, the position of the peak in the beammap can be approximated as the seafloor, and when the beam is incident at the bottom at a perpendicular angle, its peak is sharp and the detection result is Accurate [14].
Figure 5. Vertical angle incident beam pattern

The beam pattern in Figure 5 is a vertical angle incident beam pattern. It can be seen that its peak is sharp. It is more accurate to use the amplitude maximum method to take the position of its peak as the bottom area.

Figure 6. Oblique incidence angle beam diagram

When the beam is incident at an oblique incidence angle, as shown in Figure 6, its peaks are relatively flat and there will be multiple peaks. At this time, the peak position is used to take the peak position, and it is not easy to obtain the accurate bottom region position. Here, the method of amplitude maximum and sliding window is adopted to realize the detection of the bottom region. The algorithm flow is as follows:

From the previous analysis, it can be concluded that, for a vertically incident beam, the amplitude maxima method is accurate to detect, so the amplitude maxima method is first used to find the bottom position of the beam in the 128th column of the image, and denoted as T.

(2) Take the 128th column as the boundary. For the left image, from the 127th column to the 1st column, use the beam bottom position T of the i + 1 column as the window center, and then find the maximum value of the window in the window of the i column as the bottom area of the i-th column.

(3) For the image on the right, starting from column 129 to column 255, use the beam bottom position T of column i-1 as the window center, and then find the maximum value of the window in the window of column i as the position of the bottom area of column i.
(4) Record the detected bottom region positions in an array of length 255 and remove the bottom region and below.

![Figure 7](image)

**Figure 7.** It can be seen from the figure that it matches the shape of the bottom area in the image.

![Figure 8](image)

**Figure 8.** It can be seen that this method can get the bottom area more accurately.

Draw the extracted bottom location area information into a map, as shown in Figure 7. It can be seen from the figure that it matches the shape of the bottom area in the image. In order to further reflect the bottom detection result, according to the saved information, the bottom area is marked in the original image, and the bottom area is marked red. Since the detected position is a single pixel, for the display effect, the pixels near it are also performed with a mark. The results are shown in Figure 8. It can be seen that this method can get the bottom area more accurately. According to the results of bottom detection,
the data at the bottom of the pool and below are removed, and then only the data above the bottom of
the pool will be processed.

3.3. Image enhancement

Because sonar images are subject to the special characteristics of equipment and sound wave
transmission, multi-beam sounding sonar images are generally characterized by poor quality, blurred
targets, and low contrast. As can be seen from Figure 3.2 (b), Although the noise of the image is
suppressed after the median filtering process, the background gray value is very close to the target gray
value without an obvious boundary, which is not conducive to the next image segmentation operation.
Therefore, the sonar image needs to be enhanced to increase its contrast, highlight the area of interest
of the image, and weaken unwanted information. This is also the significance of image enhancement.

Commonly used image enhancement methods include contrast linear broadening, histogram
equalization, and blur domain enhancement. Due to the small difference between the gray levels of the
target and the background in the multi-beam sounding sonar image, while using the algorithm to enhance
the target area, it also increases Background area with little difference from the target gray level. Overall,
the contrast has not improved significantly.

Through further observation of the image, it can be found that the gas target area is a dense area lit a
little closer than the neighborhood. The most effective image enhancement method for this situation is
top hat processing. The principle is to subtract the image after the open operation from the original
image. The algorithm has three characteristics:

1. Highlight areas that are brighter than the area around the outline of the original image
2. Can separate plaques that are lighter than adjacent
3. Can detect the peak structure in the image

3.3.1. Image open operation. The core of top-hat processing is to open the original image first. The
open operation process is mainly divided into two steps. First, the original image is etched, and then the
image after the operation is expanded. Next, we will introduce two basic operators of corrosion and
expansion.

1. corrosion

Corrosion of an image can be approximated as the operation of finding a local minimum. From a
mathematical perspective, corrosion is the convolution of a part of the image with the kernel. The core
can be of any shape and size, and has a separately defined anchor point. The commonly used core is a
square or circle with an anchor point. The process of etching the image is shown in Figure 9. Define a
convolution kernel B, convolve B with image A, calculate the minimum value of the area covered by B,
and finally assign the minimum value to the pixel specified by the anchor point. The etch operation can
etch the edges of the image and remove "burrs."

Figure 9. Corrosion operation diagram
2. Swell
The expansion operation is a dual operation of corrosion, that is, to find a local maximum. The processing process is shown in Figure 10. Similarly define a convolution kernel B, use images A and B to convolve, calculate the maximum value of the area covered by B, and finally assign the maximum value to the pixel specified by the anchor point. The dilation operation can enlarge the edges of the image and fill out the edges of the target or internal pits.

![Figure 10. Image expansion operation](image)

The use of the corrosion operator and the expansion operator together forms an open operator. The open operation of an image can separate objects at slender points and smooth the boundaries of larger objects without significantly changing their area. Subtracting the image after the open operation from the original image can enhance the locally bright areas and suppress the background.

3.3.2. Top hat processing results analysis. In order to more intuitively verify the results of the top hat processing algorithm selected in this paper, the processing effects of several other algorithms are shown in Figure 11. The images are all images with the bottom area removed. Figure 11 (a) is the image after the contrast has been linearly expanded. It can be seen from the figure that a part of the background area is suppressed, but the background with little difference between the gray value and the target area is enhanced, and the contrast has not been significantly improved. Figure 11 (b) is an image after the histogram equalization process. Most of the background information is suppressed and the contrast has been significantly improved, but the target area has been seriously distorted after processing. Figure 11 (c) shows the result of blur domain enhancement. While enhancing the target area, the background area has been enhanced accordingly, and the processing effect is poor. Figure 11 (d) shows the processing result of the top hat. It can be seen from the figure that the target area is enhanced while the background area is suppressed, and the target area is not seriously distorted. The contrast has been significantly improved. Conducive to the next image segmentation process. In summary, the image enhancement algorithm selected in this paper is top hat processing.
The change of gray contrast can be intuitively felt from the image, and it can also be observed from the gray histogram of the image. The gray histogram of an image is a function of the gray distribution of the image. It can count the number of occurrences of different gray values. The left and right images in Figure 12 represent the grayscale histograms of the original image and the image after the top hat processing. As can be seen from the left image, the contrast between the original image target and the background is small, and it is difficult to separate the target from the background. It can be seen from the image on the right that the background information is reduced to a very low gray level, and the gray level of the target area is distributed in a relatively high area. There is a more obvious boundary between the two, which is beneficial to the The image is divided.

Figure 11. Comparison of processing effects

Figure 12. Gray histogram
3.4. Image segmentation

After the sonar image is processed by the top hat, the gray contrast is enhanced. At this time, there are mainly two pieces of information in the image. One part is foreground information that is useful for subsequent detection, and the other is useless background information. In order to separate the two pieces of information in the image, an image segmentation operation is required.

Image segmentation, as its name implies, divides a complete image into several sub-regions that make up the image. Assuming \( R \) represents the entire image area, image segmentation can be expressed as dividing the area \( R \) into \( R_1, R_2, \ldots, R_n \) sub-regions, each of which must meet the following conditions:

1. \( \bigcup_{i=1}^{n} R_i = R \)
2. \( R_i \) is a set of connected regions, \( i=1,2,\ldots,n \)
3. \( R_i \cap R_j = \emptyset \), \( i,j \in n, i \neq j \)
4. \( Q(R_i) = TRUE, i = 1,2,\ldots,n \)
5. \( Q(R_i, Y R_j) = FALSE \)

\( Q(R_i) \) refers to a logical attribute defined on a rendezvous point. The first condition states that the segmented region must be a complete segmentation, and each pixel of the image belongs to one of the subregions. The second condition is that the points in the subregion must be connected. The third condition requires that each subregion must be disjoint. The fourth condition is that the gray value of each sub-region must meet a specific attribute, for example, the gray value in each region is the same. The fifth condition states that two adjacent subregions must be different in logical attributes.

Image segmentation algorithms are mainly divided into two categories. The first is an algorithm based on the discontinuity of gray values. Segmentation is performed by finding the location of gray abrupt changes, such as edge extraction. The second is an algorithm based on the similarity of gray values, which mainly divides the image into similar sub-regions according to a predetermined rule, such as threshold segmentation. Here is the second type of algorithm. In the second class of algorithms, the algorithms with more obvious effects are the maximum entropy segmentation and the maximum inter-class variance method segmentation.

3.4.1. Maximum entropy threshold segmentation.

Entropy is a concept in information theory. It is a statistical measurement method. It is mainly used to determine the amount of information contained in random data sources. For an image \( I \), it can be considered that it contains \( N \) gray information, and each gray value is independently obtained from a limited range. The image selected in this paper is a grayscale image, \( N \) can be 256. The mathematical definition of image entropy is as follows:

\[
H(I) = -\sum_{g \in G} p(g) \log_2(p(g))
\]  

Where \( p(g) \) represents grayscale \( g \) the probability of that happening.

There are many algorithms for image segmentation using image entropy. Here, the maximum entropy segmentation method proposed by Kapupreetal is used. Given a specific threshold \( q (0 <= q < K-1) \), the image is divided into two regions \( C_0, C_1 \). The probability density function is as follows:
Among them $P_0(q)$ represents the sum of the probability of all gray values in the $C_0$ region, $P_1(q)$ the sum of the probabilities of all gray values in the $C_1$ region is added to one.

According to the definition of entropy, the entropy corresponding to the two regions can be expressed as:

\[
H_0(q) = -\sum_{i=0}^{q} \frac{p(i)}{P_0(q)} \cdot \log\left(\frac{p(i)}{P_0(q)}\right)
\]

\[
H_1(q) = -\sum_{i=q+1}^{K-1} \frac{p(i)}{P_1(q)} \cdot \log\left(\frac{p(i)}{P_1(q)}\right)
\]

For the threshold $q$, the calculated total entropy of the image is:

\[
H(q) = H_0(q) + H_1(q)
\]

Then compare the total entropy calculated by each threshold, and use the gray value at the maximum entropy as the segmentation threshold to binarize the image.

### 3.4.2. Maximum Inter-Class Variance Segmentation

Inter-class variance is a metric used in statistical discriminant analysis, and is mainly used to describe the degree of difference between the two data. In image segmentation, it can be explained that the larger the inter-class variance between the background and the target region, the greater the degree of difference, and the better the segmentation effect. This algorithm is based on a conjecture that an optimal threshold will divide the grayscale into two categories. In turn, it can be considered that if a threshold can segment the image into the best two categories from the grayscale, it can be considered as the best threshold.

The sonar image used here is a grayscale image with a total of 256 grayscale levels. It is assumed that the number of points with grayscale $i$ is $n_i$. Total number of pixels $MN = n_0 + n_1 + n_2 + ... + n_{255}$. Next, calculate the probability distribution of this image, which represents the probability of each gray value.

As with maximum entropy segmentation, set a segmentation threshold $q$ $(0 \leq q < 255)$ to divide the image into two regions $C_1$ and $C_0$.

$C_0$ The gray levels contained in are $\{0, 1, 2, ..., q\}$, $C_1$ The gray levels contained in are $\{q + 1, q + 2, ..., 255\}$. Probability of each class $\omega$ And average gray level $\mu$ As follows:

\[
\omega_0 = \sum_{i=0}^{q} p_i = \omega(q) \quad \omega_n = \sum_{i=q+1}^{255} p_i = 1 - \omega(q)
\]

\[
\mu_0 = \sum_{i=0}^{q} ip_i / \omega_0 = \mu(q) / \omega(q) \quad \mu_n = \sum_{i=q+1}^{255} ip_i
\]
Among them $\mu_i = \sum_{i=0}^{255} ip_i$ represents the average gray level of the entire image. The definition of the variance between classes can be given according to the following formula:

$$\sigma^2 = \omega_0(\mu_0 - \mu_i)^2 + \omega_1(\mu_1 - \mu_i)^2$$

$$= \omega_0 \omega_1 (\mu_1 - \mu_0)^2$$  \hspace{1cm} (12)

Calculate the inter-class variance corresponding to each gray level, obtain the gray value when the inter-class variance is the largest, and use this as the segmentation threshold to divide the image into two categories.

3.4.3. **Comparative analysis of segmentation results.** The effect of the two types of segmentation is shown in Figure 13. Figure a is an image formed by using the largest inter-class variance method. You can see that its foreground information contains many small areas. There is more sub-information, and its segmentation effect is poor, which is not conducive to the extraction of suspected areas of gas leakage in the next step. Figure b is the result of the maximum entropy segmentation algorithm. It can be seen from the figure that the foreground information mainly includes suspected gas leakage areas and some smaller areas. There is less misclassified information, and the segmentation effect is good, which can reduce difficulty in extracting suspected areas of gas leakage. To sum up, the last segmentation algorithm selected in this paper is maximum entropy segmentation.

![Figure 13. Comparison of segmentation results](image)

3.5. **Acquisition of Suspected Gas Leaks**

After the image is segmented, the image information can be mainly divided into foreground information and background information, and the suspected gas region to be extracted exists in the foreground information. The next work is to extract the suspected gas leakage area and make subsequent judgments. The extraction steps are as follows:

1. Open the image, fill the holes, smooth the outline of the object, break the narrow connections, and eliminate the small protrusions.

2. The connected area is marked, and the area is labeled for each area to calculate the area, and the largest area is taken as the suspected gas leakage area.
3.5.1. Connected region marker. The segmented image is called a binary image and contains only two gray values of 0 and 255. The most commonly used method for counting the size of each region in a binary image is connected region labeling. In the image, the adjacent areas with the same gray value are called connected areas. The connected area labeling is to mark each individual connected area as a block by marking the white area in the binary image. Area statistics can be performed on these blocks.

There are two basic definitions of connected areas, 4 adjacency and 8 adjacency. The schematic diagram is shown in Figure 14.

![Figure 14. Schematic diagram of 4 adjacency and 8 adjacency](image)

Set its center element \( f(x, y) \). The definition of 4 adjacency considers that the four points of up, down, left and right are connected, while the 8 adjacency thinks that in addition to the up, down, left, and right, points on the diagonal can also be considered connected to it. In image processing, if only a rough label is required, then the 4 adjacency rule is used, and if a more accurate connected area label is used, an adjacency rule is used. Through Figure 15, the difference between the 4 adjacency criterion and the 8 adjacency criterion can be understood more intuitively. Using the 4 adjacency criterion, three connected regions are marked. With the 8 adjacency criterion, only two regions can be marked. The squares in the figure are four pixel values. The connected region labeling criterion chosen in this paper is the 8 adjacency criterion.

![Figure 15. Schematic diagram of Unicom area marking](image)

3.5.2. Analysis of Suspected Area Extraction Results. Suspicious region extraction is mainly divided into two steps. Figure 16 shows the result map of each step and the final extraction result. From left to right are the segmentation result map, binary image open operation result map, Adjacent connected area labeling results map, gas suspected leakage area result map. From the data point of view, a total of 40 connected areas are labeled. The gray level of each connected area is assigned according to its label. In order to show the effect, a pseudo-color processing is performed on the labeled image. From the final extraction result graph, it can be seen that some disturbing small areas and bottom areas are completely removed, and only one connected area is left. This area can be initially considered as a suspected gas leakage area. To finally determine whether there is a gas leakage, you need Subsequent estimation of the gas rising form.
3.6. Gas rise pattern estimation

After the processing in the previous section, the suspected gas leakage area in the image was extracted, and further analysis is needed to determine if there is a gas leak. Prof. Douglas has conducted a plume analysis of the leaked gas. It is found that when a pipeline located on the sea floor leaks, the gas will have a clear upward trend. By estimating the rising form of the gas, it can be judged whether it is a leaked gas[14]. The simplest and most effective method for estimating the rising form of gas in the image is to extract the linear trend of the target area using Hough transform, and determine the straight line shape of the area by determining the angle between the straight line and the coordinate axis[30]. From the analysis in Section 2.1, it can be known that the scene actually detected by the multi-beam sounding sonar is a fan-shaped area, but it will be displayed in a rectangular manner during imaging, which will cause the straight lines in the actual scene to be curved in the rectangular image of. In order to restore the real situation and ensure the accuracy of the detection results, interpolation processing is needed to restore the actual fan display according to the rectangular information and beam angle information of the image, and then Huff transform is performed on the fan-shaped image to extract straight lines.

3.6.1. Coordinate transformation and image interpolation. Multi-beam sounding sonars will emit a set of fan beams during actual imaging, and then return the value of each beam, so when stored and displayed, it will be a rectangular image with the abscissa being the beam number and the ordinate being the sample value. In order to restore the actual fan display, the new image size after conversion needs to be estimated first. The image resolution used in this paper is $255 \times 1500$, that is, there are 255 beams, each beam has 1500 samples, and the angle of each beam is $0.5^\circ$. Based on this information, the size of the converted image can be estimated. The height of the new image is the same as the height of the original image, and the width can be given by Equation 2-13:

\[
\text{col} = \text{round}(2 \times \sin(\frac{63.5}{180} \times \pi) \times \text{row})
\] (13)
The height of the new image row is 1500. After inserting the formula, the width of the new image can be obtained as 2685. After determining the size of the new image, the gray value in the new image needs to be restored according to the information of the original image. Here, this article is selected from the most classic interpolation algorithms based on backward mapping. The algorithm idea of backward mapping is to inverse transform the coordinates in the transformed image and then find its position in the original image. The principle is shown in Figure 18.

According to the data format used in this paper, for each position \((x, y)\) in the fan-shaped image, the inverse transformation formula is as follows, where \(c_{\text{pos}}\) is the beam number, which is the column number in the original image, and \(r_{\text{pos}}\) is the sampling time, which is the line number in the original image. The angle \(\sigma\) is calculated by the arctangent function.

\[
c_{\text{pos}} = \frac{\sigma - 26.5}{0.5} + 1 \tag{14}
\]

\[
r_{\text{pos}} = \text{round}\left(\sqrt{\left((x - 1)^2 + (y - \frac{\text{col}}{2})^2\right)^0.5}\right) \tag{15}
\]

The beam number and sampling position obtained according to the inverse transformation formula may not be integers. At this time, an interpolation algorithm is needed to solve the gray value of the position. Commonly used interpolation algorithms mainly include nearest neighbor interpolation algorithms and bilinear interpolation algorithms.

The idea of the nearest neighbor interpolation algorithm is to find the gray value of the four points closest to the target point in the original image, and then set the weighting coefficient by distance and distance, and weight the four points to obtain the gray value of the corresponding point in the fan chart. This algorithm is simple to implement, the algorithm runs fast, and the interpolation image is shown in Figure 19.

The idea of the bilinear interpolation algorithm is also to find the nearest four points in the original image, and then perform a linear operation in the horizontal and vertical directions. Based on the results of the linear operation, a fitting is performed to obtain the gray values of the corresponding points in the fan diagram. This algorithm has a good imaging effect and is generally used for data visualization display, but the calculation speed is slow and it is not suitable for embedded implementation. The interpolation image is shown in Figure 20.

![Figure 17. Backward Mapping](image1.png)

![Figure 18. Nearest Neighbor Interpolation Algorithm](image2.png)
3.6.2. Ascension pattern estimation based on Hough transform. The new image obtained after interpolation transformation can realistically restore the actual underwater situation. Next, the rising image of the gas leakage area needs to be estimated on the new image. In this paper, we use the Hough transform to extract the linear features of the target area to determine whether it is a rising pattern.

The algorithm idea of Hough transform is to obtain a set of a specific shape by calculating a local maximum of the accumulated results in a parameter space. For the detection of straight lines, a transformation between two coordinate systems is usually used to map a straight line in a rectangular coordinate system to a point in a polar coordinate system to form a peak, thereby converting the problem of detecting a straight line into a statistical peak problem.

A straight line in the rectangular coordinate system can be used $y = k_0x + b_0$ Means, among them $k_0$ is the slope of the line, $b_0$ is the intercept of the line, $k_0$ and $b_0$ It can be considered as a point in the parameter space $(k, b)$. However, the slope is infinite when the line is perpendicular to the x-axis, which is not easy to calculate. To solve this problem, Duda and Hart proposed the use of coordinate systems $(\rho, \theta)$. To represent the parameters of a straight line, the expression is as follows:

$$\rho = x \cos \theta + y \sin \theta$$  \hspace{1cm} (16)

$r$ Indicates the distance from the origin to the straight line, $\theta$ is $\rho$ and x-axis Angle. Use $(\rho, \theta)$ Space can effectively solve the problem of $(k, b)$ space divergence. After the above transformation formula, the straight lines in the original image can be mapped as $(\rho, \theta)$ A point in space. After that, you only need to count the accumulation times of each point to get the straight line length in the original image. And can be based on $\theta$ The value gives the angle between the straight line and the x-axis as well as the y-axis. The coordinate transformation diagram is shown in Figure 21.
In order to determine the upward trend of the target area, first the N longest candidate straight line set in the target area is extracted by Huff transform, and then the angle between each line and the y-axis is obtained, and the average of the N angles is used as the regional main axis rising angle. When the rising angle satisfies \((-30^\circ, 30^\circ)\), it can be considered that the target area has a rising tendency and there is a gas leak.

3.6.3. Line extraction and analysis of detection results. Figure 3.18 is the image of the target area after interpolation and one of the straight lines extracted from the target area. The width of each straight line in the image is a single pixel. Extracting a single straight line is not enough to represent the main axis direction of the entire area. In order to calculate the results accurately, a total of three straight lines are extracted here. \(-13^\circ, -14^\circ\). By calculating the average angle of the three straight lines to be \(-13.3^\circ\), the rising angle of the main axis of the target area is determined to be \(-13.3^\circ\), which meets the requirements of \((-30^\circ, 30^\circ)\). It is believed that there is a gas leak in this area. The extracted three straight lines are marked on the graph, as shown in Figure 3.19. From the figure, it can be intuitively seen that the extracted straight lines conform to the rising pattern of the entire area.
Figure 22. Line mark

By using the above algorithm to detect 120 sonar images obtained in the pool, 80 pictures contain gas leaks and 40 pictures do not contain gas leaks. It is possible to accurately detect the presence or absence of gas in 118 of the images, and the detection accuracy rate reaching 98.3%, it can be proved that the algorithm designed can effectively judge whether there is a gas leak in the sonar image.

4. Conclusion
Firstly, the multi-beam sounding sonar image is analyzed, and a processing flow is designed according to its image characteristics. Then the effects of different algorithms in the processing flow are compared. Finally, a set of gas leak detection algorithms suitable for the multi-beam sounding sonar image is determined. The main algorithms of the corresponding process are: median filtering, amplitude maxima plus sliding window bottom detection, top hat processing, maximum entropy segmentation, 8 adjacent connected region labeling, nearest neighbor interpolation algorithm based on backward mapping, Huff transform-based ascending pattern estimates. According to the designed algorithm, 120 pictures were detected, and the detection accuracy was 98.3%. It can be verified that this algorithm can detect the gas leak in the sonar image more accurately.

References
[1] Du Fei. Research on detection method of underwater gas leak based on bubble acoustics [D]. Tianjin: Doctoral Dissertation of Tianjin University, 2016. 1-3
[2] Wei Lei. Analysis and Optimization of Jinghui Pipeline Leak Detection System [D]. Xi'an: Master's Degree Thesis of Xi'an Shiyou University, 2011. 1-4
[3] Miao Chengzhang. Submarine landslide and its impact on submarine pipelines [D]. Zhejiang: Master's Degree Thesis of Zhejiang University, 2007. 1-3
[4] Fang Na, Chen Guoming, Zhu Hongwei, Meng Huixing. Statistical analysis of submarine pipeline leakage accidents [J]. Oil & Gas Storage and Transportation, 2014, 33 (1): 99-102
[5] Zhao Huijun, Wu Weiqiang, Wang Kehua, et al. Detection and Location of Oil and Gas Pipeline Leakage Based on Infrasound Method [J]. Oil & Gas Storage and Transportation, 2012 (03): 65-68.
[6] Dong Liewu, Zhong Jiayong. Application of domestic SCADA system in East China Product Oil Pipeline Control Center [J]. Oil Depot and Gas Station (5): 34-38.
[7] Wang Xueqiang, Su Xin, Yang Wei. Overview of Oil and Gas Pipeline Leak Detection Technology [J]. NATURAL GAS & OIL, 2007, 25 (3): 19-23.
[8] Yue Junhong, Wang Tao, Ren Yingqiao, et al. Oilfield Oil and Gas Pipeline Leak Detection Based on Neural Network Algorithm [J]. Electronic Design Engineering, 2017 (10).
[9] Lei Hulin, Cao Qiu'e, Liu Yijun, Ma Chaosong, Xu Xingping. Ding Gaoxiang. Research on
computer software based oil and gas pipeline leak detection technology [J]. Yunnan Chemical Industry, 2019 (12): 1-2.

[10] J Schneider von, Deimling J Brockhoff J Greinert. Flare imaging with multibeam systems: Data processing for bubble detection at seeps [J]. AGU and the Geochemical Society, 2007, 8(6): 1-7

[11] Aneta Nikolovska, Heiko Sahling, Gerhard Bohrmann. Hydroacoustic methodology for detection, localization, and quantification of gas bubbles rising from the seafloor at gas seeps from the eastern Black Sea [J]. AGU and the Geochemical Society, 2008, 9(10): 1-13

[12] Thomas C. Weber, Larry A. Mayer, Jonathan Beaudoin. Mapping Gas Seeps with the Deepwater Multibeam Echosounder on Okeanos Explorer [J]. CENTER FOR COASTAL AND OCEAN MAPPING, 2012, 25(1): 54-55

[13] Schneider von Deimling, C Papenberg. Technical Note: Detection of gas bubble leakage via correlation of water column multibeam images [J]. Ocean Science, 2012, (8): 175-181

[14] Douglas S Wilson, Ira Leifer, Eric Maillard. Megaplume bubble process visualization by 3D multibeam sonar mapping [J]. Marine and Petroleum Geology, 2015, (68): 753-765

[15] Peter Urban, Kevin Koser, Jens Greinert. Processing of multibeam water column image data for automated bubble/seep detection and repeated mapping [J]. Limnology and Oceanography: Methods, 2016, (00): 1-15

[16] Pu Ding, Zhang Wanyuan, Guo Jun, et al. Multi-beam sonar imaging and detection algorithm for underwater gas targets [J]. Applied Science and Technology, 2017, v.44; No.294 (05): 16-20.

[17] John E Hughes Clarke. Applications of multibeam water column imaging for hydrographic survey [J]. Hydrographic Journal, 2006, 120(4): 3-11.

[18] Yuan Yanyi, Liu Xiao, Xu Chao, Li Haisen. Underwater imaging technology based on multibeam sounding system [J]. Marine Surveying and Mapping, 2012, 32 (4): 29-30.

[19] Shen Zhengyan. Research on sonar image denoising and segmentation technology [C]. Harbin: PhD dissertation of Harbin Engineering University, 2010. 1-2.

[20] Tang Chunrui. Multi-resolution analysis and recognition of underwater target acoustic image [C]. Harbin: Harbin Engineering University, 2009. 30-35

[21] Gonzalez R C, Woods R E, Digital Image Processing [M]. Translated by Ruan Qiuqi et al. Beijing: Electronics Industry Press, 2007.

[22] Zong Siguang, Wang Jiang’an. Multi-scale morphological operator fusion image filtering technology and filtering quality evaluation [J]. Chinese Journal of Optics, 2005 (09): 26-30.