

## Article

# Contrast-Invariant Edge Detection: A Methodological Advance in Medical Image Analysis

Dang Li , Patrick Cheong-Iao Pang \* and Chi-Kin Lam 

Faculty of Applied Sciences, Macao Polytechnic University, Macao 999078, China; p2209276@mpu.edu.mo (D.L.); cklamsta@mpu.edu.mo (C.-K.L.)

\* Correspondence: mail@patrickpang.net

**Abstract:** Edge detection methods are significant in medical imaging-assisted diagnosis. However, existing methods based on grayscale gradient computation still need to be optimized in practicality, especially in terms of actual visual quality and sensitivity to image contrast. To optimize the visualization and enhance the robustness of contrast changes, we propose the Contrast Invariant Edge Detection (CIED) method. CIED combines Gaussian filtering and morphological processing methods to preprocess medical images. It utilizes the three Most Significant Bit (MSB) planes and binary images to detect and extract significant edge information. Each bit plane is used to detect edges in  $3 \times 3$  blocks by the proposed algorithm, and then the edge information from each plane is fused to obtain an edge image. This method is generalized to common types of images. Since CIED is based on binary bit planes and eliminates complex pixel operations, it is faster and more efficient. In addition, CIED is insensitive to changes in image contrast, making it more flexible in its application. To comprehensively evaluate the performance of CIED, we develop a medical image dataset and conduct edge image and contrast evaluation experiments based on these images. The results show that the average precision of CIED is 0.408, the average recall is 0.917, and the average F1-score is 0.550. The results indicate that CIED is not only more practical in terms of visual effects but also robust in terms of contrast invariance. The comparison results with other methods also confirm the advantages of CIED. This study provides a novel approach for edge detection within medical images.



Academic Editor: Fabio La Foresta

Received: 5 December 2024

Revised: 13 January 2025

Accepted: 16 January 2025

Published: 19 January 2025

**Citation:** Li, D.; Pang, P.C.-I.; Lam, C.-K. Contrast-Invariant Edge Detection: A Methodological Advance in Medical Image Analysis. *Appl. Sci.* **2025**, *15*, 963. <https://doi.org/10.3390/app15020963>

**Copyright:** © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** medical image; edge detection; bit plane; contrast invariance

## 1. Introduction

Edge detection plays a vital role in image processing. Its aim is to identify the boundaries or edges in images, which is indispensable for object recognition, image segmentation, and feature extraction [1]. Edge detection methods have been widely applied in multiple fields including computer vision, industrial inspection, and medical imaging. Although they share the common goal of identifying boundaries, the characteristics of the images in different fields pose unique challenges.

In the field of image edge detection, there are many significant differences between medical image edge detection and natural image edge detection [2]. In terms of detection goals, medical image edge detection is mainly used for precisely locating lesion areas, and segmenting different tissue types, its core objective is to provide support for assisting in diagnosis and treatment planning [3]. In contrast, natural image edge detection is mainly applied to object recognition, contour extraction, and scene analysis. It aims to help computers understand the main objects and structures in a scene so as to support

various computer vision applications. From the perspective of detection requirements, when medical images are acquired by medical equipment, noise and artifacts are likely to be generated. This requires that medical image edge detection adopt methods that can effectively suppress noise and artifacts while not losing important edge information. Moreover, the grayscale distribution of medical images is narrow and the contrast is low. The gray values of different tissues and organs may overlap, and the grayscale changes in diseased tissues may be very subtle. Therefore, the edge detection algorithms for medical images need to be sensitive to subtle grayscale changes in order to accurately detect the edges of diseased tissues. Natural images contain rich colors and textures, and the shapes of objects and backgrounds are diverse, so their edge detection needs to be able to distinguish the boundaries between different objects as well as those between objects and backgrounds. Regarding the demand for accuracy, medical image edge detection requires a high level of accuracy in its results. Incorrect edge detection may lead to misdiagnosis, and the detected edges must be as close as possible to the actual boundaries of tissue structures. However, the results of natural image edge detection are allowed to have some errors to a certain extent [4]. In non-critical applications such as simple image recognition or contour-based image retrieval, even if there are small deviations in the edge detection results, as long as they can roughly reflect the shapes of objects, the requirements can still be met. In terms of real-time requirements, medical image edge detection is usually carried out in an offline state, and there is not a high requirement for real-time performance. However, natural image edge detection needs to be processed in real time in applications such as autonomous driving, which imposes relatively high requirements on the computational efficiency and response speed of algorithms.

Given the imaging characteristics of medical images, such as low contrast and narrow gray scale regions, as well as their high requirement for the accuracy of edge detection, we focus our research on the field of medical image edge detection. The aim is to accurately extract the edge information with important diagnostic value so as to provide doctors with clearer and more accurate diagnostic assistance in imaging.

In the process of rapid development of modern medicine, medical image-assisted diagnosis has become an indispensable and important means. Medical images generally include images obtained through various imaging technologies, and common medical images include X-ray, computed tomography (CT), magnetic resonance imaging (MRI), ultrasound imaging, and nuclear medicine imaging [5–7]. Medical images enable doctors to intuitively analyze the internal organ structure of the body, thus assisting them in diagnosing diseases and formulating treatment plans. In the process of medical image-assisted diagnosis, edge detection is a key technology: the precision and reliability of its edge detection are crucial in the whole diagnostic process [8,9].

Accurate, complete, and stable medical image edge detection results can help doctors quickly and accurately locate the lesion area and identify key information such as perimeter, dimensions, and form of the lesion [10]. This helps doctors to not only make quick diagnostic decisions but also provide an accurate basis for subsequent treatment plans. For example, in the diagnosis of tumor diseases, edge detection can determine the growth site and range of the tumor, providing an important reference for the scope of surgical resection, to avoid excessive resection or incomplete resection.

Traditional edge detection methods mainly rely on the first- and second-order derivatives of the image and identify the edges in the image by detecting the changes in the gray values. The Roberts operator uses local differences based on first-order derivatives to detect edges. It calculates the gradient magnitude for edge detection by utilizing the difference between two adjacent pixels in the diagonal direction. The Roberts operator is highly sensitive to noise and lacks directional information [11]. Prewitt and Sobel operators are both

based on first-order derivatives [12]. The Prewitt operator measures the gradient using two  $3 \times 3$  convolution kernels, which are used to calculate the horizontal and vertical gradients separately. The structure of the Sobel operator and the Prewitt operator is similar, the difference being that it has different weights. Prewitt and Sobel are simple and effective but sensitive to noise [13]. The Laplacian operator is a second-order derivative-based method that determines edges by detecting changes in the second-order derivatives, and it is often combined with Gaussian filtering to detect edges due to it being very sensitive to noise [14]. The Canny operator is among the more commonly employed methods. The Canny initially conducts Gaussian smoothing on the image for noise reduction. Subsequently, it calculates the gradient magnitude and direction of each pixel. Finally, it acquires the ultimate edges through non-maximum suppression and double thresholding [15]. The Canny operator employs the first-order derivatives of an isotropic Gaussian kernel and is regarded as the theoretically ideal way for detecting isolated edges contaminated by additive white Gaussian noise [1].

In recent years, in the medical field, scholars have proposed many different methods of edge detection for the characteristics of medical images. Rajan et al. [16] proposed an edge detection method based on Gaussian gradient for retinal Optical Coherence Tomography (OCT) images. They utilized the convolution result of a Gaussian function and its first-order derivative as a kernel, and the method was able to efficiently extract the boundary information in retinal OCT images. Mittal et al. [17] proposed a productive edge detection technique B-Edge. The B-Edge algorithm calculates the gray threshold and adjusts the intensity. It adopts a triple-intensity threshold automatic selection method to cover the gray range and determines the edge pixels through horizontal and vertical scanning, thus improving edge connectivity. Hien et al. [18] proposed an MRI edge detection method based on Semi-Translation Invariant Contourlet Transform (STICT) and Fuzzy C-Means (FCM) clustering, and finally, the Canny operator is used to recognize the edges. The method performs well in improving image quality and edge detection accuracy. Nikolic et al. [19] adapted the improved Canny operator for medical ultrasound images. They replaced the Gaussian filter with an adapted median filter and a weighted smoothing filter. This substitution aimed to diminish the impact of speckle noise on edge detection and enhance the accuracy of edge detection. To further remove the interference of Salt-and-Pepper noise as well as random noise in the image, Topno et al. [20] put forward an edge detection approach based on median filtering, which better retains the edge information and does not destroy the detail information, can effectively detect the edges in medical, natural or industrial images, and has strong versatility. Elmi et al. [21] proposed an edge detection method founded on the matching tracking algorithm for multiple application scenarios. The algorithm transforms the edge detection issue into a signal processing matter. It holds the merits of being noise-insensitive and capable of detecting weak edge pixels. It shows potential for application in the diagnosis and treatment of medical image-related diseases. Lin et al. [22] proposed a quasi-high-pass filtering operator for medical images, which calculates the local grayscale mean value and local signal energy variations within a  $3 \times 3$  neighborhood. The operator has good adaptivity and isotropic symmetry, and it can accurately locate the edges and reduce the blurring of the edges.

However, owing to the intricacy and particularity of medical images, these methods may not be able to accurately recognize the edges of the lesions when the medical image contrast is low, resulting in incomplete or inaccurate edge information being extracted. To overcome these problems, we propose the Contrast-Invariant Edge Detection (CIED) method, which does not directly rely on the change of grayscale but combines the information of three Most Significant Bit (MSB) planes to obtain the final edge detection result.

The CIED is capable of efficiently extracting the image edges and is insensitive to different contrasts.

The contributions of this paper are summarized and listed below:

- A new Contrast-Invariant Edge Detection (CIED) algorithm is proposed. It performs better in terms of visualization and applicability.
- Experimental results are presented to confirm the effectiveness of CIED. This includes visualization, as well as contrast-invariant robustness and comparison with other methods.
- A new edge detection test dataset based on medical images is created. It contains different kinds of medical images that can be efficiently employed to assess the performance of edge detection methods.

This paper's primary structure is as follows. Section 2 introduces the concepts of Most Significant Bit (MSB), Least Significant Bit (LSB), and bit plane. Section 3 describes the CIED method. Section 4 shows the detailed experimental results and comparisons. Section 5 further analyzes the experimental results and discusses the CIED method. Section 6 summarizes our findings.

## 2. Preliminaries

### 2.1. The Most Significant Bit (MSB) and The Least Significant Bit (LSB)

Digital image processing and many aspects of related fields are highly dependent on the MSB and the LSB [23]. For a pixel value in a digital image, it usually consists of multiple bits. For an 8-bit grayscale image, MSB refers to the corresponding binary value in the highest bit with a weight of  $2^7$ . As shown in Figure 1, the pixel value is 149, and MSB is distributed in the leftmost of the 8 bits with a value of one. The MSB contributes more to the pixel value and it largely determines the main features and contours of the image. This is due to a change in the higher bits causing pixel values to be highly variable. In the visualization of an image, the information represented by the MSB is usually expressed in terms of the overall lightness and darkness and the main structure of the image. For example, changing the MSB may cause the image to change from bright to dark, or may lead to significant changes in the main contours. When it comes to image processing and analysis, MSB is crucial. In image compression, MSB usually contains important information about the image, so in some compression algorithms, MSB is protected to ensure that key image features are not lost in the compression process [24]. In image enhancement and filtering processes, the manipulation of the MSB can effectively adjust the contrast and brightness of images and highlight the main structure of images [25]. In image segmentation, MSB is usually used for threshold-based segmentation methods [26]. By choosing a suitable threshold to determine the MSB of a pixel value, an image can be quickly divided into different regions. This MSB-based threshold segmentation method is computationally efficient and can quickly provide approximate segmentation results of an image. It can provide an initial segment region for subsequent more accurate segmentation algorithms. In noise removal and detail preservation, MSB can be used to localize noise [27]. Due to the stability of the MSB, it is an important basis for determining whether a pixel has been disturbed by noise. Noise leads to random alterations in the lower bits of pixel values. When the MSB of a certain pixel varies considerably from those of its neighboring pixels and its lower bits change randomly, it is highly likely to be affected by noise. In this case, the pixel is corrected according to the MSB of the surrounding pixels to effectively remove the noise.



**Figure 1.** The demonstration of MSB and LSB with a pixel value of 149.

The LSB is the bit with the lowest weight. For an 8-bit binary pixel value, the LSB is the corresponding binary value in the lowest bit with weight  $2^0$ . As shown in Figure 1, the pixel value is 149, and LSB is distributed in the rightmost of the 8 bits with a value of one. The LSB contributes less to the pixel value, but it contains information about the details and small changes. In the grayscale image, LSB variations may show up as subtle textures and noise in the image. Since the variation of LSB has less impact on the overall visual effect of the image, it is often used in some special image processing methods like digital watermarking and steganography. By modifying the LSB, information can be hidden without causing significant visual changes [28]. The LSB is relatively sensitive to noise and interference. This is due to it having a low weight, and small variations may result in a change in its value. However, it is also due to this sensitivity of the LSB that it can be used to detect and analyze image changes in some specific applications [29]. For example, in image tampering detection, the changes in LSB can be analyzed to determine whether the image has been modified.

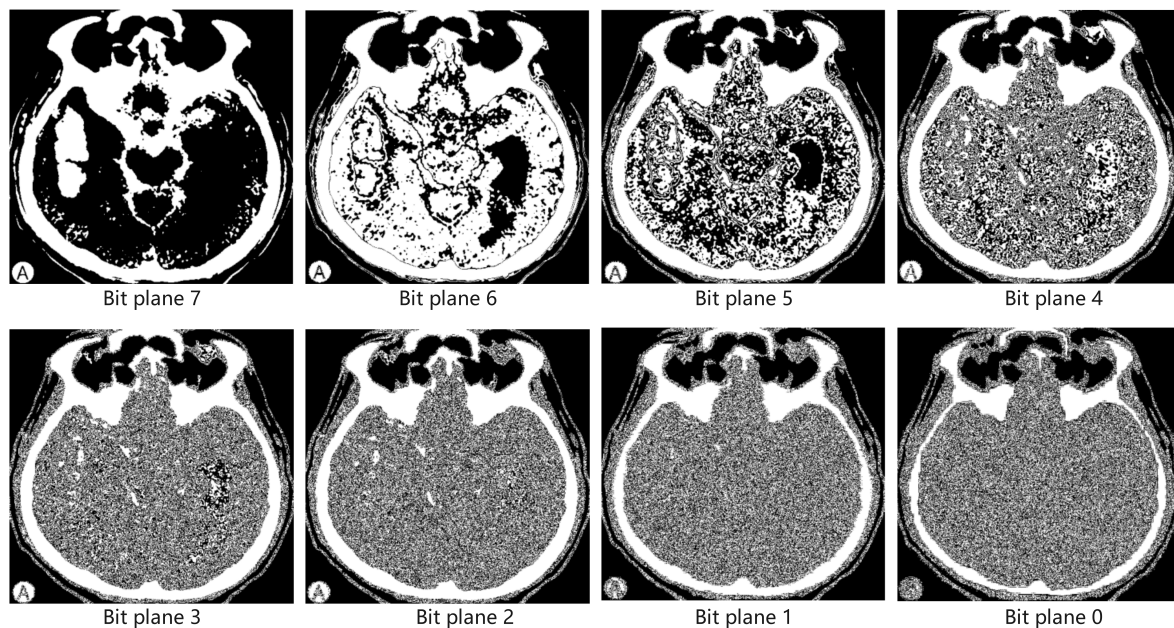
## 2.2. Bit Plane

Bit planes occupy an important position in digital image processing. They are multiple planes formed by separating the grayscale pixel values by bit [30]. A typical 8-bit grayscale image can be split into 8-bit planes. Suppose that pixel  $P$  of an 8-bit grayscale image can be represented as Equation (1), where  $b_i (i = 0, 1, 2, \dots, 7)$  is a binary digit with a value of either zero or one.

$$P = \sum_{i=0}^7 b_i \cdot 2^i \quad (1)$$

For example, the binary representation of a pixel value of 206 is 11001110, which has a value of one in Bit Plane 7, one in Bit Plane 6, ..., and zero in Bit Plane 0.  $P$  performs a bit plane decomposition into eight binary bit planes  $b_7, b_6, \dots, b_0$ , each bit plane  $b_i$  is formed by the  $i$ th bit of all pixels in the image. To visualize the information distribution in each bit plane more intuitively, we chose a CT image of the brain and decomposed it into 8-bit planes. Figure 2 shows the image information contained in each bit plane from high to low. It can be visualized that the high-bit planes hold a majority of the contour and structural information. The low-bit plane is barely visible as edges and contours and contains more detail and noise. The properties of the bit plane have led to extensive applications across multiple image processing domains. In image compression, image size can be reduced by selective compression based on the importance of the bit planes [31]. In image enhancement, the processing of the special localization plane allows targeted adjustment of the contrast, brightness, and other features of the image [32]. In image encryption, the transformation of specific bit planes enables secure protection of the image content by encrypting the pixel values within those planes using advanced cryptographic algorithms [33]. In digital watermarking, by selecting appropriate positions and methods to embed watermark information in bit planes, the function of copyright protection can be achieved without interfering with the normal visual effects of the image [34]. Usually, the lower bit planes are chosen to embed watermarks because the slight changes in the lower bit planes have no significant impact on the visual quality of the image. Overall,

the analysis of the bit plane helps to deeply understand the inner structure of the image, providing new perspectives and methods for image processing.



**Figure 2.** Bit-plane decomposition of the brain CT image.

### 3. Proposed Methods

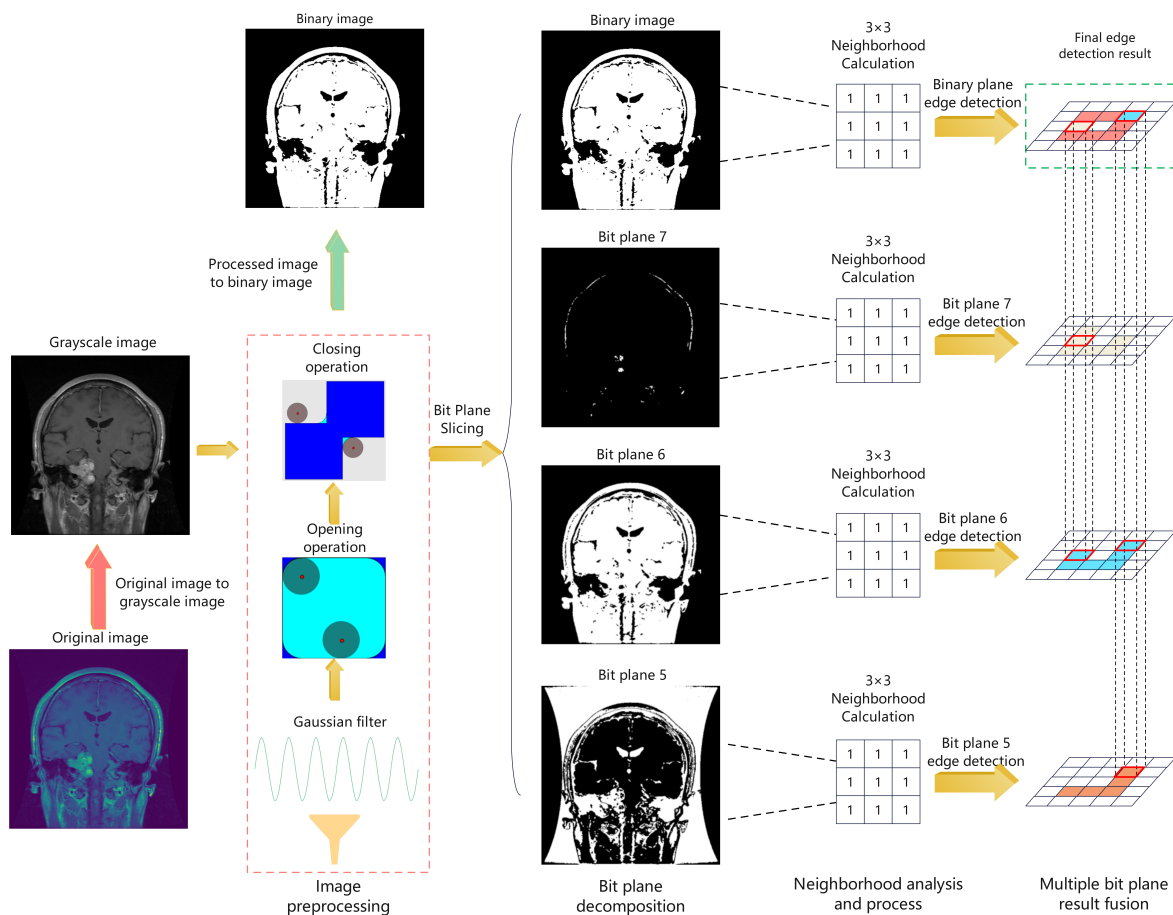
Medical image-assisted diagnosis is crucial for disease diagnosis and treatment, and edge detection methods play a significant role in it. Many existing edge detection techniques rely mainly on the calculation of the gray-scale gradient, which performs in medical images with low contrast and blurred edges not well and struggles to meet the diagnostic accuracy and efficiency requirements. To solve the above problems, we propose the CIED method. The CIED can extract clear and complete edges on medical images of different contrasts with less noise interference.

The grayscale image's pixels can be regarded as binary values ranging from 0 to 255. A normal grayscale image consists of 8-bit planes. The three MSB planes of the image are rich in the main contour information and have a natural advantage for extracting the image edges, while the five LSB planes are rich in the details of the image and are not conducive to highlighting the global structure of the image. Based on the properties of bit planes, the CIED takes advantage of the three MSB planes. Figure 3 illustrates the demonstration of the CIED. The CIED includes four main steps: image preprocessing, grayscale image bit plane decomposition,  $3 \times 3$  neighborhood analysis and process, and edge detection result fusion.

If the original image to be detected is a color image, its conversion to a grayscale image is necessary. In the image preprocessing stage, the method combining Gaussian filtering and morphological processing is adopted to process the original grayscale image. First, we use Gaussian filtering to process the image. This can effectively reduce Gaussian noise, make the image smoother, and better preserve the edge and detailed information of tissues and organs in medical images. Then, the combination of morphological opening and closing operations can further process the image. The opening operation can remove small bright noise points and to a certain extent smooth the contours of objects. The closing operation can connect the broken parts of the edges of objects and fill in small holes, making the edges of objects more complete. This processing procedure can fully utilize the advantages of the two methods. It can better highlight and optimize the important

structures and edges in medical images while reducing noise. After preprocessing, we can obtain the processed image.

The processed image is decomposed into 8-bit planes, and 3 MSB planes are selected. As shown in Figure 3, the three MSB planes are Bit Plane 7, Bit Plane 6, and Bit Plane 5. These bit planes contain almost all the edge information. Bit Plane 7 tends to highlight the most significant, overall contour information of the main objects or regions in the image, presenting a relatively rough but clear edge outline. Bit Plane 6 tends to show more detailed edge profile information than Bit Plane 7 and starts to show edges of relatively large localized features on the basis of the main profile. Bit Plane 5 contains a bit more rich edge detail. In addition to the major and larger localized edge contours covered in the previous two planes, there are also some smaller localized edge variations and other information. At the same time, we use the Otsu method to obtain the binary image. The binary image obtained after processing by Otsu’s method can divide the original image pixels into the background part and the foreground part by automatically determining the threshold value, which can separate the target from the background and highlight the main features [35]. As shown in Figure 3, the binary image is capable of outlining the general outline of a grayscale image, which also contains the relatively simple and more obvious edge information in the grayscale image. Here, the binary image can be considered as a layer of the bit plane. There are four bit planes in total, including 3 MSB planes and a binary image.

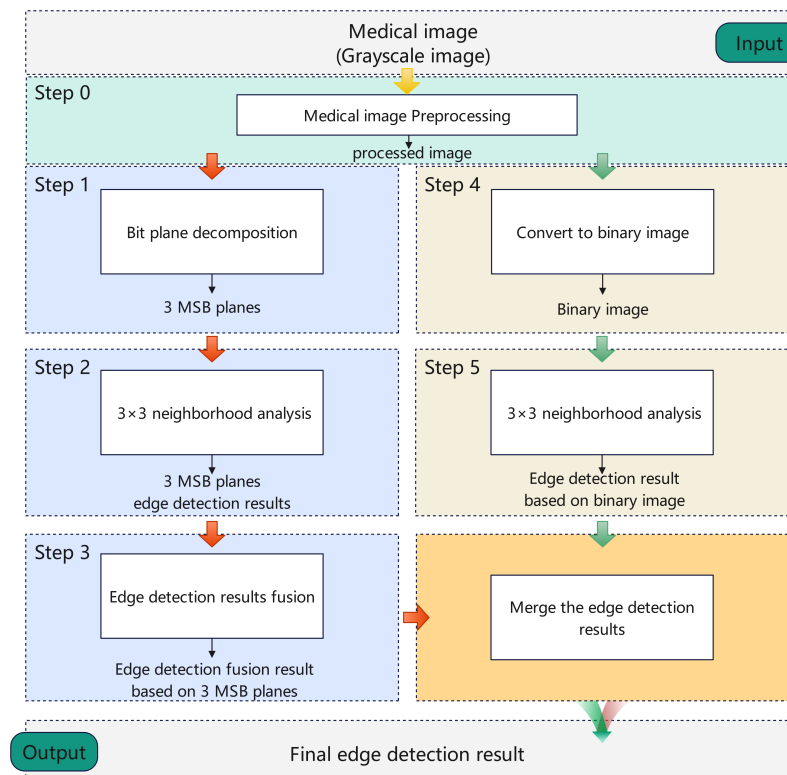


**Figure 3.** The demonstration of the proposed Contrast-Invariant Edge Detection (CIED) method.

We perform the neighborhood analysis process on these 4 layers of bit planes, respectively, to obtain the edge detection result. For a better grasp of the local characteristics of the image, we choose the 3 × 3 neighborhood to analyze the bit plane. Compared with larger

size neighborhoods,  $3 \times 3$  neighborhoods involve fewer pixels, and  $3 \times 3$  neighborhoods can effectively capture information in eight different directions. According to the sum of neighboring pixel points, we judge whether the pixel point is an edge point. The neighborhood analysis step does not depend on a specific direction, and as long as the threshold condition is satisfied in the  $3 \times 3$  neighborhood, it is determined as an edge point, so it has the advantage of isotropy. When performing  $3 \times 3$  neighborhood analysis on the bit plane, in order to improve computational efficiency, we adopt the integral image method to perform integral operations on the entire bit plane. Then, the sum of the  $3 \times 3$  neighborhood of any pixel can be obtained through simple addition and subtraction operations using the values at four positions of the integral image. It can avoid the repeated summation operation of the nine pixels in the  $3 \times 3$  neighborhood one by one, improving the calculation speed. After the neighborhood analysis step, each bit plane obtains an edge detection result. Next, the edge detection results of the three MSB planes are fused. When the sum of the values of the corresponding pixels within the 3 MSB planes reaches 2 or exceeds it, a value of 1 is assigned to the empty plane, thereby attaining a fused outcome for edge detection. Finally, the edge detection outcomes of the binary image and the fused edge detection results of the 3 MSB planes are combined to obtain the ultimate edge detection result.

Figure 4 shows the flowchart of the proposed CIED method. The flowchart is divided into 6 main steps, where Step 0 is preprocessing of the grayscale image, Steps 1 to 3 on the left side are processed for three MSB planes, and Steps 4 to 5 on the right side are processed for the binary image. The red and green arrows indicate the order in which the steps are executed, and the small arrow in the box indicates the output of the current step. In fact, the processing of 3 MSB planes and the binary image are synchronized in order. Here, we describe them separately for easier observation and understanding. The detailed CIED edge detection process is explained below.



**Figure 4.** The flowchart of the proposed Contrast-Invariant Edge Detection (CIED) method.

- **Input:** The input should be grayscale images. In the case of a color image, it should be converted into a grayscale image for uniform processing. We employ the weighted



average method in line with the human eye's sensitivity to the distinct colors of red, green, and blue to assign different weights to each channel, and then the pixel values of the three channels are weighted and summed according to the corresponding weights. The result is the pixel value of the converted grayscale image. For an original color image, its red component is set as  $R_o$ , its green component as  $G_o$ , and its blue component as  $B_o$ . The grayscale value  $I$  can be calculated by the following Equation (2), where coefficients 0.299, 0.587, and 0.114 are based on the sensitivity of the human eye to the colors red, green, and blue. In practical applications, it is widely used in the field of image processing and it can well convert color images into grayscale images that conform to the visual perception of the human eye.

$$I(x, y) = 0.299R_o(x, y) + 0.587G_o(x, y) + 0.114B_o(x, y) \quad (2)$$

- Step 0: This step is image preprocessing. First, Gaussian filtering is performed on the image. Gaussian filtering is a type of linear smoothing filter. For each pixel in an image, its new value is determined by the weighted average of the pixels in its neighborhood. The weights are determined by the Gaussian function. The farther a pixel is from the central pixel, the smaller its weight. In this way, Gaussian noise in the image can be effectively reduced while keeping the edges and details of the image relatively clear. For a two-dimensional image, the Gaussian function can be expressed as Equation (3), where  $(x, y)$  is the coordinate relative to the center pixel and  $\sigma$  is the standard deviation, which controls the width of the distribution of the Gaussian function. It is common to use a convolution kernel  $K$  of finite size, whose elements  $K(i, j)$  are computed by the Gaussian function  $G(x, y)$  and normalized so that all the elements sum to 1. The input grayscale image is  $I(x, y)$ , and the filtered image  $J(x, y)$  is obtained by Equation (4). After Gaussian filtering processing, we perform the opening operation and the closing operation on the image successively. The opening operation involves performing an erosion operation on the image first and then a dilation operation. The formula for the opening operation is shown in Equation (5), where  $J$  is set to be the Gaussian filtered image and  $E$  is the structural element. The erosion operation  $J \ominus E$  shrinks the bright objects in the image and removes some small bright noise points or thin connecting parts. Then, the dilation operation  $J \oplus E$  restores the size of the objects to a certain extent, but it does not restore the small objects or noise points that have been eroded away, thus achieving the effect of removing small bright noise points and smoothing the contours of the objects. The closing operation first performs a dilation operation on the image and then an erosion operation. The formula for the closing operation is shown in Equation (6). The dilation operation  $J \oplus E$  reduces the holes in the dark objects in the image and connect the broken parts. Then, the erosion operation  $J \ominus E$  restores the approximate original size of the objects, but it does not restore the small holes filled by the dilation or the broken connected parts, thus achieving the purpose of connecting the broken parts at the edges of the objects and filling the small holes. By the method of Gaussian filtering followed by morphological processing, we obtain the processed image. Gaussian filtering is performed first to remove noise, and then morphological opening and closing operations are processed to further optimize the structure and edge information of the image on the basis of reducing noise interference, improving the quality of the image, and facilitating the subsequent image analysis and processing.

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3)$$

$$J(x, y) = \sum_{i=-1}^1 \sum_{j=-1}^1 I(x+i, y+j)K(i, j) \tag{4}$$

$$J \circ E = (J \ominus E) \oplus E \tag{5}$$

$$J \bullet E = (J \oplus E) \ominus E \tag{6}$$

- Step 1: This step is bit-plane decomposition. There is a grayscale image of 8-bit depth with each pixel taking values ranging from 0 to 255, and each bit plane is a binary image composed of only 0 and 1. If  $x, y$  denote the coordinates of the processed image  $I_p$  and  $B_i$  denotes the binary value on the  $i$ th bit plane of that pixel value, then the pixel value at the  $x, y$  position can be represented as Equation (7). The main contour information contained in the 3 MSB planes is relatively stable and does not change significantly due to lighting variations and imaging angles. As the 3 MSB planes encompass the principal structure and contour details of the image and exhibit lower sensitivity to noise, the CIED method employs these 3 MSB planes for detecting the edge of the medical image. Consequently, a bit-plane decomposition of the grayscale image is carried out. Each pixel value is divided into different bit planes, and then we extract the 3 MSB planes using the following Equation (8). After bit-plane decomposition, we obtain the 3 MSB planes of the grayscale image.

$$I_p(x, y) = \sum_{i=0}^7 B_i(x, y) \cdot 2^i \tag{7}$$

$$B_i(x, y) = \frac{I_p(x, y) \& 2^i}{2^i}, \quad i \in \{7, 6, 5\} \tag{8}$$

- Step 2: This step is the  $3 \times 3$  neighborhood analysis and process on 3 MSB planes. After obtaining the 3 MSB planes, we perform the  $3 \times 3$  neighborhood analysis and process for the 3 MSB planes to obtain the edge detection result of each plane. The  $3 \times 3$  neighborhood can fully consider the local information around a pixel to more accurately identify whether it is an edge point. To obtain more efficient edge detection results, we utilize the integral image technique to accelerate the processing procedure of the  $3 \times 3$  neighborhood. First, we calculate the integral image of each entire bit plane. For each pixel point  $(x, y)$  on the bit plane, the integral image of the bit plane  $I_b(x, y)$  is calculated by Equation (9). Here,  $B_i(m, n)$  represents the pixel value at position  $(m, n)$  on the bit plane. We add up all the pixel values in the rectangular area from the upper left corner to position  $(x, y)$  to obtain integral image  $I_b(x, y)$  of the entire bit plane. During this calculation process, in order to calculate the integral image of the bit plane more quickly, we can use Equation (10), where  $I_b(x-1, y)$  represents the integral sum of the rectangular area to the left of the current point,  $I_b(x, y-1)$  represents the integral sum of the rectangular area above the current point, and  $I_q(x-1, y-1)$  is the part that has been repeatedly calculated in the upper left corner and needs to be subtracted. Finally, by adding the pixel value  $B_i(x, y)$  of the current point, the value of the current point  $(x, y)$  in the integral image is obtained. In this way, through Equation (10), the integral value corresponding to each pixel point in the integral image can be efficiently calculated without having to recalculate the sum of the pixels in the rectangular area starting from the upper left corner every time. Second, we use the calculated integral image to determine the sum of pixel values in the  $3 \times 3$  neighborhood of each pixel. For the pixel point at the position  $(x, y)$ , the sum of pixel values in its  $3 \times 3$  neighborhood  $I_b(x, y)$  can be calculated by Equation (11). This method of calculating the sum of the  $3 \times 3$  neighborhood using the integral image improves the speed compared to calculating the pixel values in the

$3 \times 3$  neighborhood one by one. Because it only needs to perform simple addition and subtraction operations on the values of four specific positions in the integral image to obtain the sum of the  $3 \times 3$  neighborhood, avoiding the repeated summation operation of the 9 pixels in the  $3 \times 3$  neighborhood one by one, and improving the computational efficiency. Based on the sum of the neighborhood  $S_i(x, y)$ , we determine whether the pixel is an edge point according to the Equation (12). Here,  $B'_i(x, y)$  represents the edge detection result of the bit plane. If the sum of neighborhood  $S_i(x, y)$  of a pixel is less than or equal to 2 or greater than or equal to 7, then the pixel is set to 0 which is the background, otherwise it is set to 1 which is the edge. When the sum of the values in the  $3 \times 3$  neighborhood is 0 or 9, the current pixel point is regarded as the background. When the sum of the values in the neighborhood is greater than 1 and lesser than 8, the current pixel point is set as an edge point. Such a judgment condition is relatively loose, which identifies too many edge points and thus results in thicker edges. When the sum of the values in the neighborhood is greater than 3 and lesser than 6, the current pixel point is set as an edge point. Such a judgment condition is relatively strict and sometimes may lead to the loss of some edge points. Therefore, in this paper, we choose 2 and 7 as the thresholds to determine whether a pixel point is an edge point or not. The rule can effectively identify edge pixel points, removing isolated noisy pixels to some extent, while also ensuring edge connectivity. In addition, the  $3 \times 3$  neighborhood has the same orientation sensitivity whether it is horizontal, vertical, or diagonal 45-degree orientation, so it has the advantage of isotropic. By performing the  $3 \times 3$  neighborhood analysis and processing on 3 MSB planes, we can obtain the edge detection results of 3 MSB planes, and each MSB plane obtains an edge detection result.

$$I_b(x, y) = \sum_{m=0}^x \sum_{n=0}^y B_i(m, n) \tag{9}$$

$$I_b(x, y) = I_b(x - 1, y) + I_b(x, y - 1) - I_b(x - 1, y - 1) + B_i(x, y) \tag{10}$$

$$S_i(x, y) = I_b(x + 1, y + 1) - I_b(x - 1, y + 1) - I_b(x + 1, y - 1) + I_b(x - 1, y - 1) \tag{11}$$

$$B'_i(x, y) = \begin{cases} 0, & \text{if } S_i(x, y) \leq 2 \text{ or } S_i(x, y) \geq 7 \\ 1, & \text{otherwise} \end{cases} \tag{12}$$

- Step 3: This step is edge detection results fuse. By combining the information from the 3 MSB planes, richer image features can be obtained. Different bit planes contain different levels of image information, and by fusing them, a richer and more accurate edge description can be obtained. For example, in brain CT images, some edges may not be very obvious on a single-bit plane. However, by fusing the information of the 3 MSB planes, the edge details of lesions can be clearly presented, providing a more comprehensive diagnostic basis for doctors. Based on the edge detection results of 3 MSB planes, we create an empty template  $B'_i(x, y)$  with 0 value. As shown in Equation (13), if the sum of the corresponding position in each bit plane of 3 MSB planes reaches 2 or exceeds it, the corresponding position of the template is assigned to 1, otherwise it remains 0. If the threshold is set to 1, the judgment condition is relatively loose, and thus a small amount of noise from the lower bit planes is introduced. If the threshold is set to 3, the judgment condition is too strict and some fused edge points are lost. Here, we set the threshold to 2. The edge information of the 3 MSB planes can be effectively fused while a small number of noise points can be filtered out at the same time. This process ensures that the edge information of 3 MSB planes is fused and compensated for each other, and some false edges are filtered out to produce a more comprehensive

edge result. By performing Edge detection results fuse on 3 MSB planes, we can obtain the fused edge detection result of the 3 MSB planes.

$$B'_i(x, y) = \begin{cases} 1, & \text{if } \sum_{i=5}^7 B'_i(x, y) \geq 2 \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

- Step 4: This step directly generates a binary image from the grayscale image of the original medical image. At the same time, the grayscale image of the original image is used as input to further improve the continuity and accuracy of the edge detection. The Otsu algorithm is also used to acquire binary images in order to further improve the continuity and accuracy of edge detection. The Otsu algorithm is capable of automatically calculating an optimal threshold according to the grayscale histogram of the image, thereby partitioning it into foreground and background segments and generating a binary image. The value of a binary image is the same as a bit plane: it only contains 0 or 1 values. Essentially, it can be considered as a bit plane. By performing this step, we can obtain a binary image of the grayscale image.
- Step 5: This step is the  $3 \times 3$  neighborhood analysis and process on the binary image. Same as Step 2, we also perform the  $3 \times 3$  neighborhood analysis and process on the binary image. As shown in Equation (14), where  $B(x, y)$  denotes the binary image, we can compute the integral image  $I_q(x, y)$  of the binary image. Similarly, we can use the recursive Equation (15) to accelerate the process of calculating the integral image. As shown in Equation (16), we can obtain the sum of  $3 \times 3$  neighborhoods by integral image directly. Finally, we can obtain the edge detection result  $B'_b(x, y)$  by Equation (17).

$$I_q(x, y) = \sum_{m=0}^x \sum_{n=0}^y B(m, n) \quad (14)$$

$$I_b(x, y) = I_q(x - 1, y) + I_q(x, y - 1) - I_q(x - 1, y - 1) + B(x, y) \quad (15)$$

$$S(x, y) = I_q(x + 1, y + 1) - I_q(x - 1, y + 1) - I_q(x + 1, y - 1) + I_q(x - 1, y - 1) \quad (16)$$

$$B'_b(x, y) = \begin{cases} 0, & \text{if } S(x, y) \leq 2 \text{ or } S(x, y) \geq 7 \\ 1, & \text{otherwise} \end{cases} \quad (17)$$

- Output: Finally, according to the binary image edge detection result and the 3 MSB plane fused edge detection result, we combine these two results to obtain the ultimate edge detection result. As Equation (18) shows, the ultimate edge detection result  $U(x, y)$  is obtained by taking the union of the edge detection result of the binary image  $B'_b(x, y)$  with the edge detection result of the 3 MSB planes  $B'_i(x, y)$ .

$$U(x, y) = \begin{cases} 1, & \text{if } B'_b(x, y) = 1 \text{ or } B'_i(x, y) = 1 \\ 0, & \text{otherwise} \end{cases} \quad (18)$$

In the proposed CIED method, we utilize 3 MSB planes to extract edge information. The  $3 \times 3$  neighborhood processing is performed for each plane to obtain the edge detection results of each plane, and then the results of the 3 planes are compensated and fused to obtain the fused edge detection result. To further improve the clarity and the continuity of the edge line segments, we also employ the  $3 \times 3$  neighborhood to process the binary image converted from the grayscale image and thereby obtain the edge detection results. The final edge detection output is derived by merging the two edge detection results. With these steps, CIED is able to effectively extract the edges of medical images. In contrast to other edge detection approaches, the CIED process is founded on the bit plane, which solely comprises 0 or 1. The bit plane has a smaller amount of data and simpler arithmetic

logic than dealing with grayscale values from 0 to 255. Consequently, the CIED not only improves computational performance but also reduces computational complexity.

#### 4. Experimental Results and Analysis

Due to the unique characteristic of contrast invariance in the CIED, it aims to deal with different types of medical images under different contrast conditions. The existing public medical image datasets can hardly cover such diverse and targeted image samples comprehensively, and thus cannot fully validate the effect of this characteristic. Therefore, to comprehensively assess the performance of the CIED, we develop a Medical Image Edge Detection Test (MIEDT) dataset. The MIEDT includes 100 medical images randomly chosen from three publicly available datasets: Head CT-hemorrhage [36], Coronary Artery Diseases DataSet (<https://www.kaggle.com/datasets/younesselbrag/coronary-artery-diseases-dataset-normal-abnormal> (accessed on 6 January 2023)), and Skin Cancer MNIST: HAM10000 [37]. In addition, we label the ground truth (GT) with the assistance of experienced physicians. We make many modifications based on the physician's recommendations. Through repeated confirmations, we finally obtain the GT that meets the actual needs. The MIEDT consists of 15 head CT images, 25 coronary artery disease images, and 60 skin lesion images. In MIEDT, the head CT images demonstrate high contrast due to the significant difference in the absorption of X-rays into tissues of different densities. The coronary artery images are acquired by CT angiography, which has low contrast, as it is characterized by subtle tissue differences. The skin images rely on optical imaging and exhibit the lowest contrast because they capture surface features and color changes. Since these medical images have different contrast and imaging modalities, the MIEDT has the capacity to comprehensively assess the performance of the edge detection algorithm. The MIEDT has been released on Kaggle (<https://www.kaggle.com/datasets/lidang78/miedt-dataset> (accessed on 12 January 2025)). The MIEDT is continuously updated and maintained.

To objectively measure the performance of the edge detection methods, we utilize metrics like accuracy, precision, recall, and F1-score [1,38]. In this paper, we conduct a comprehensive evaluation of edge detection results, which is divided into three main parts: performance evaluation of the CIED, the contrast-invariant robustness of the CIED, and comparisons. All experiments and analysis are implemented in Python 3.8 on a computer equipped with an Intel(R) Core(TM) i5-10200H CPU, 16 GB RAM, and an NVIDIA GeForce GTX 1050 Ti graphics card.

##### 4.1. Evaluation Metrics

In the task of edge detection, in addition to intuitive visual assessment, it is crucial to assess edge detection methods' performance in an objective and comprehensive manner. To evaluate edge detection algorithms from different perspectives, we employ evaluation metrics such as accuracy, precision, recall, and F1-score, which collectively provide a comprehensive assessment of the performance of edge detection algorithms from various dimensions [39].

Accuracy reflects the ratio of pixels that are correctly classified, specifically the percentage of pixels correctly identified as either edge or non-edge. The formula is defined as Equation (19). True Positives ( $TP$ ) refer to pixels correctly identified as edges. True Negatives ( $TN$ ) indicate pixels correctly recognized as non-edges. False Positives ( $FP$ ) represent pixels incorrectly labeled as edges. False Negatives ( $FN$ ) denote pixels that are actually edges but were missed. However, in edge detection tasks, general edge pixels

usually occupy a small proportion relative to non-edge pixels, and relying only on accuracy may lead to less comprehensive results.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (19)$$

Precision is used to measure how many of the edges detected by the algorithm are true edge pixels. It is calculated by Equation (20). High precision means that the edge detection algorithm produces fewer false edges and the detected edges are usually accurate.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (20)$$

Recall is employed to measure the ability to recognize edges that are actually present, and the formula is shown in Equation (21). High recall means that most of the actual edges are recognized and fewer edges are missed.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (21)$$

The F1-score combines the roles of precision and recall, and it is used as a composite evaluation when both are equally important. Thus, it can provide a balanced view between the accuracy and completeness of the detection results. The formula of the F1-score is shown in Equation (22).

$$\text{F1-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (22)$$

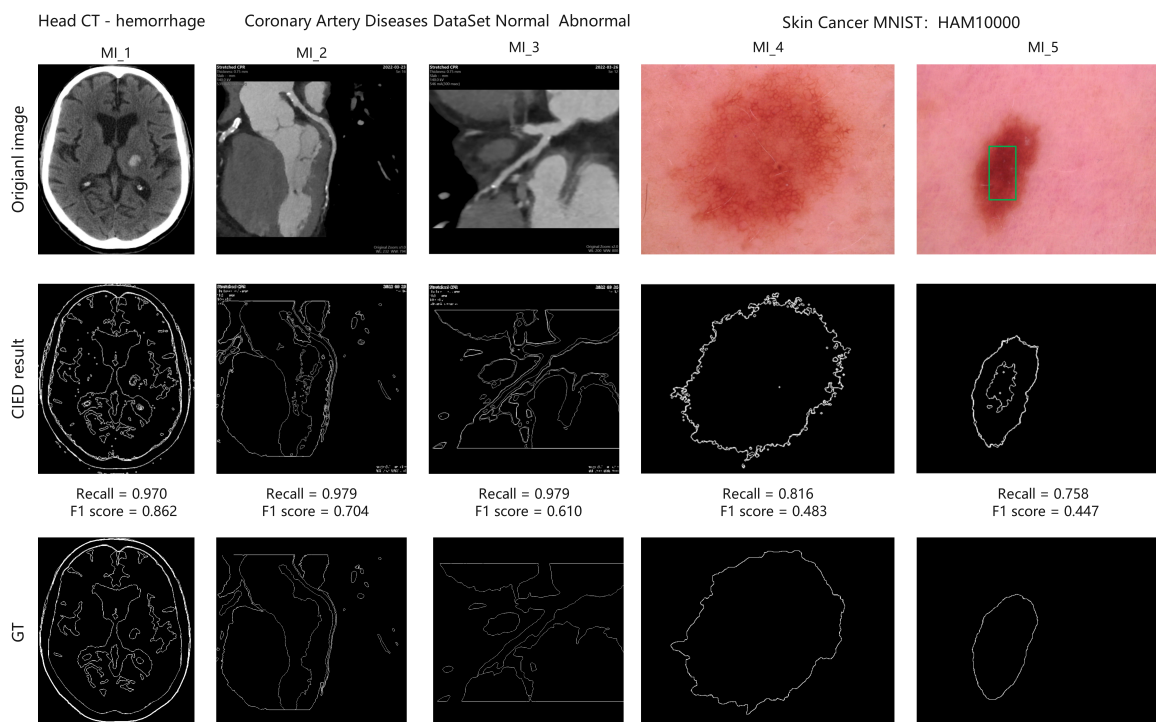
By using these four evaluation metrics, the performance of edge detection methods can be evaluated from different perspectives in medical image application scenarios.

#### 4.2. Performance Evaluation of the CIED

We evaluate the edge detection method using medical images from the MIEDT dataset. The MIEDT dataset has played a crucial role in this research, and its data content has provided support for edge detection methods. For the convenience of research and use, this dataset can be accessed through Kaggle. The specific steps are as follows: First, we visit the address corresponding to the dataset (<https://www.kaggle.com/datasets/lidang78/miedt-dataset> (accessed on 12 January 2025)), and the dataset can be obtained through the download operation. After obtaining the dataset, according to the detailed instructions, we can further understand the structure of the dataset, the data format, and how to use code to load the data and other information.

To evaluate the intuitive visual effort, we use the CIED method for edge detection on medical images. Partial results are shown in Figure 5. The first row MI\_1 to MI\_5 is the original medical images, the second row is the detected edge images of CIED, and the third row is the GT of the medical images. In addition, Figure 5 also shows the recall and F1-score of the edge detection results. We can intuitively observe that CIED can extract complete and clear edges stably even though these images come from different types and have different contrast distributions. From the detection results of MI\_1 in Figure 5, the CIED can extract the boundary of different tissues and structures of the brain completely. From the detection results of MI\_2 and MI\_3 in Figure 5, CIED can extract the complete heart contour and can accurately detect the contour edges of the coronary arteries. As a result of MI\_4 and MI\_5 in Figure 5, CIED can accurately extract the edges of skin lesions and meanwhile capture their subtle features. For example, as shown in the results of MI\_5 in Figure 5, the detected edge in the green box further targets the core lesion area. In addition, recall is high in all edge detection results, indicating that CIED has a high edge

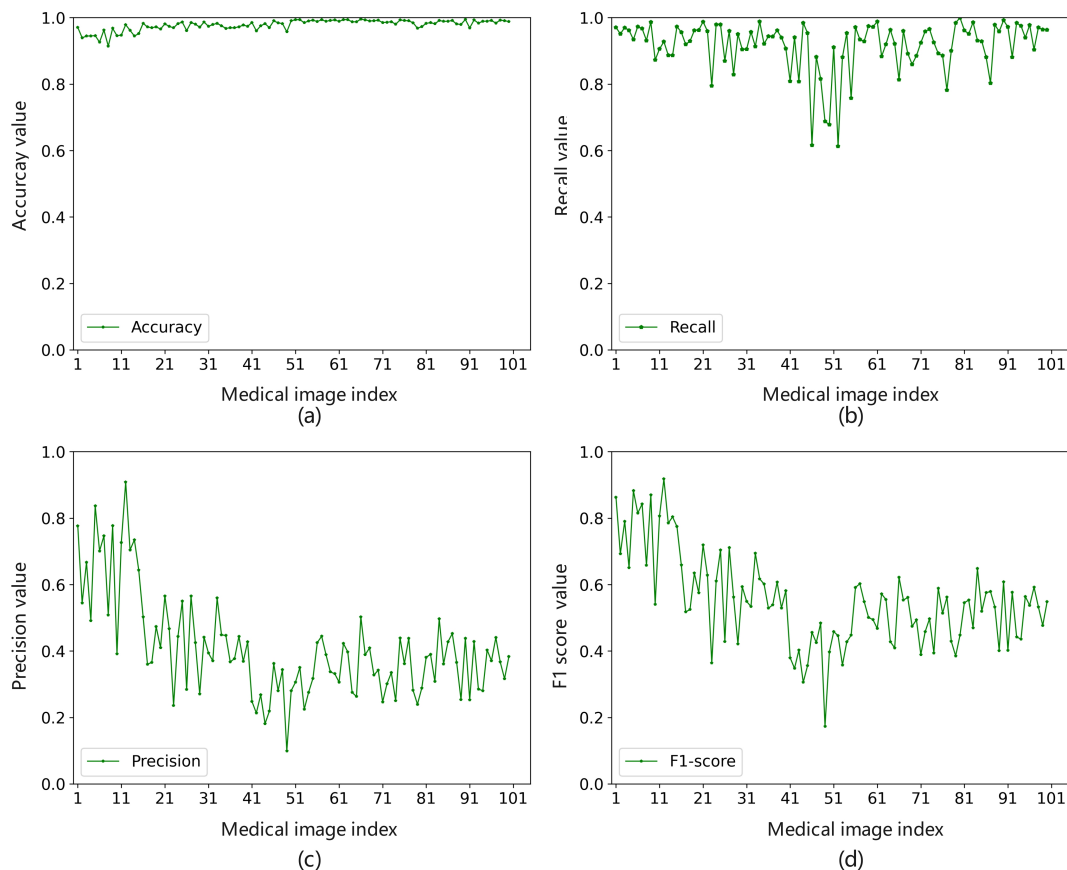
detection rate. Overall, in terms of visual effect, CIED can not only effectively extract the key contours in medical images, but also extract the edges with slight variations, and these edges can further help doctors target the potential lesion areas.



**Figure 5.** The demonstration of detected edge images using proposed CIED in different images. The green box indicates the potential edges of the skin lesion.

To further quantitatively evaluate the CIED, we record the evaluation results of edge detection with four evaluation metrics including accuracy, precision, recall, and F1-score. Figure 6 demonstrates the performance of CIED under each evaluation metric. Figure 6a shows the accuracy of the CIED method. It stays at a high level with an average value of 0.978. This indicates that the edge pixels obtained by CIED have a relatively high classification accuracy. However, since the number of edge pixels is generally small, relying on the accuracy alone is not comprehensive. Figure 6b shows that the recall of the CIED method performs well. It can detect most of the real edge information with an average value as high as 0.917. This indicates that CIED can effectively detect edge information in different types of medical images. The CIED is highly effective in detecting most of the real edge information in medical images, hardly missing any important edge details. Figure 6c indicates that the precision of CIED method performance is mediocre on some medical images, where the mean value is 0.408. This is because the CIED extracts edge information as comprehensively and accurately as possible to prevent missed diagnoses in medical diagnosis. In clinical practice, the complete and accurate delineation of the edges of the lesion area is crucial for accurate diagnosis and the formulation of subsequent treatment plans. Even the omission of tiny edge details may lead to missed diagnoses, which then affects the treatment outcomes and prognoses of patients. Therefore, CIED tends to obtain rich and complete edge information to reduce the risk of missed diagnoses. However, in cases where the number of real edge pixels in some medical images is scarce, the precision rate shows a relatively high sensitivity. For example, MI\_4 and MI\_5 in Figure 5, the number of their real edge pixels is relatively small. Even if CIED extracts a small number of potential edges, the proportion of these potential edges in the overall detection results is

relatively significant, which then leads to a decline in precision rate. Figure 6d shows the F1-score performance, and it is more similar to precision due to the high value of recall, with the mean value of 0.550. Taken together, CIED is well suited for medical images. It can effectively detect edge information in medical images. The CIED can detect all potential edges of the lesion areas as much as possible, avoiding missed diagnoses.



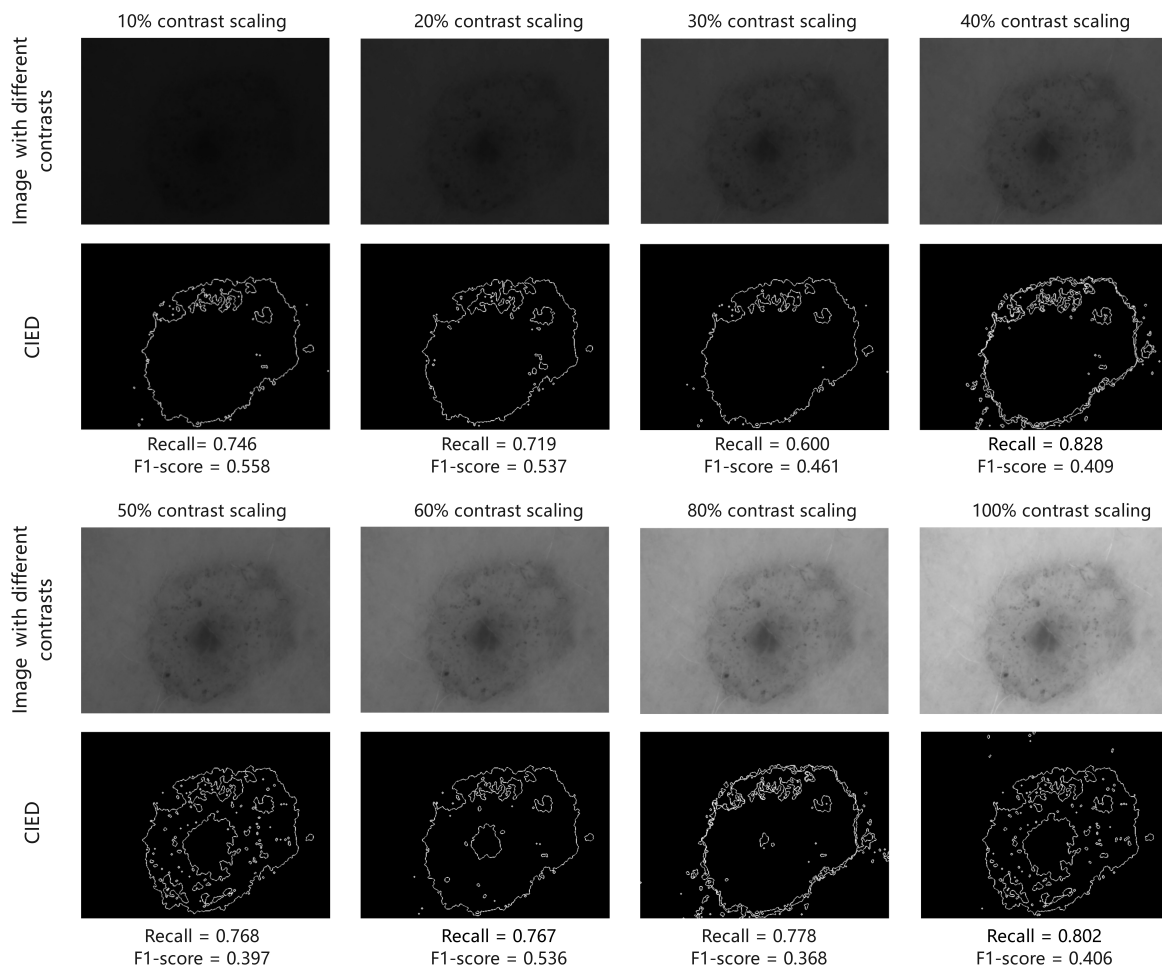
**Figure 6.** The results of edge images in terms of accuracy, precision, recall, and F1-score. (a) The accuracy result. (b) The recall result. (c) The precision result. (d) The F1-score result.

#### 4.3. The Contrast-Invariant Robustness of the CIED

In addition to visual and quantitative evaluations, we also evaluated the contrast-invariant robustness of CIED. To verify the contrast-invariant of CIED, we linearly scaled the contrast of the image to different degrees, and then we used CIED for these images separately. Figure 7 shows the visual effect of the CIED at different contrasts from 10% to 100%. Rows 1, 3 of Figure 7 are medical images at different contrasts and Rows 2, 4 of Figure 7 are CIED edge detection results at different contrasts. In addition, Figure 7 shows the recall and F1-score at various contrast levels. From Rows 1 and 3 of Figure 7, it can be observed that when the contrast is only 10%, the image is barely visible. As the contrast gradually increases, the image becomes clearer and clearer. From Rows 2, 4 of Figure 7, it can be observed that CIED can identify edges clearly even when the contrast of the image is only 10%. With the gradual increase in contrast, CIED's edge detection visualization has been relatively stable and able to extract complete and clear edges. From the result of Figure 7, the recall of CIED is always high, almost always above 0.7. This indicates that CIED can extract the edges effectively in different contrasts. Especially in the case of low contrast, for example, in the contrast of only 10%, its edge detection results of recall can also reach 0.746. Due to the low contrast instead of reducing part of the noise interference, the performance of the F1-score reaches 0.558.



In addition, we also recorded detailed evaluation metrics for CIED at different contrast levels from 10% to 100%. Table 1 shows the specific evaluation metrics results for CIED at different contrast levels. We can observe that the CIED is able to maintain a high recall at any contrast. The CIED mainly utilizes the three MSB planes' information of the image to determine edges. When performing linear contrast transformation, the brightness distribution and color perception of the image change. Due to the change in contrast, regions in the image that originally had similar brightness may have new edges appear as the brightness difference increases, or regions that originally belonged to edges may have their edges weakened or even disappear as the brightness difference decreases. From the perspective of pixel values, their distribution is readjusted according to the contrast transformation rules, and the pixel distribution of each bit plane also changes accordingly. In addition, the edge and texture information of the image is also affected. Originally distinct edges may become indistinct, and the clarity and distinguishability of textures also change depending on the contrast. Even when the contrast changes significantly, CIED can still capture the key edge information in the image relatively accurately. This phenomenon further illustrates that the CIED is robust in the sense that it is contrast-invariant, and in the case of low contrast it also performs well.



**Figure 7.** The demonstration of detected edge images using proposed CIED at different contrast levels.

**Table 1.** The result of detected edge images at different contrast levels in terms of accuracy, precision, recall, and F1-score.

Evaluation Metrics	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	Average
Accuracy	0.985	0.984	0.983	0.972	0.975	0.984	0.972	0.969	0.964	0.975	0.977
Precision	0.445	0.428	0.374	0.271	0.267	0.412	0.246	0.241	0.198	0.271	0.316
Recall	0.746	0.719	0.600	0.828	0.768	0.767	0.809	0.778	0.794	0.802	0.761
F1-score	0.558	0.537	0.461	0.409	0.397	0.536	0.378	0.368	0.318	0.406	0.437

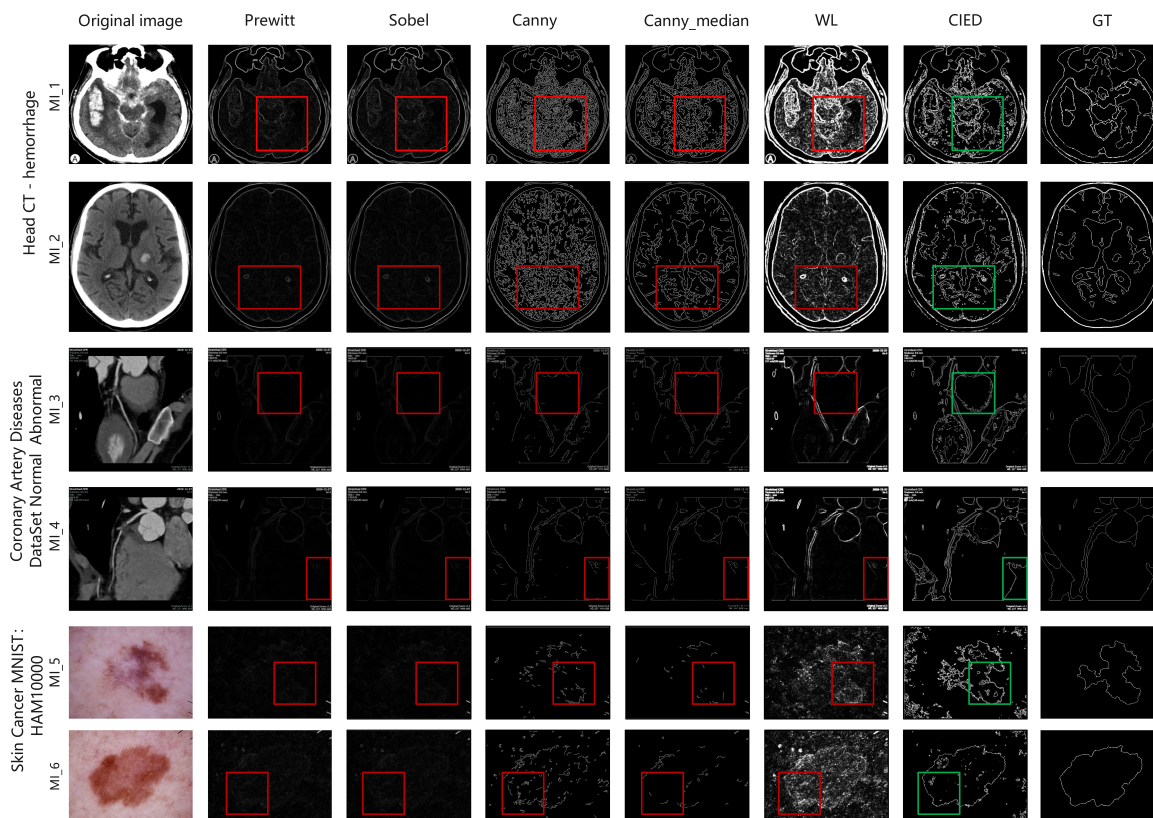
#### 4.4. Comparison

To further evaluate the advantages of the CIED, we compare it with other edge detection methods, which include three traditional edge detection operators Prewitt [40], Sobel [41], and Canny [15], and two improved edge detection algorithms, Canny–Median [20] and the WL operator [22]. We evaluate these methods in three ways: intuitive visual evaluation, quantitative evaluation, and contrast-invariant robustness comparison.

To intuitively compare the visual effects of different edge detection methods, we present some of the medical edge images for different types and contrasts. Figure 8 shows that Column 1 is the original image of MI\_1 to MI\_6, Columns 2 to 7 are the edge detection results of Prewitt, Sobel, Canny, Canny\_median, WL, and CIED, respectively, and the last columns are the GT of the original images. Each row corresponds to the detected edge result of a specific image using different methods. In the first row MI\_1 and second row MI\_2 of Figure 8, Prewitt and Sobel can extract edges, except for the region with weak contrast, and miss some edges. Canny and Canny–Median can extract valid edges, but at the same time, many false edges are also extracted. WL and the CIED can distinguish the main edge information, and the CIED is visually better. The edge marked by green rectangles has more complete information and less noise. In the third row MI\_3 and fourth row MI\_4 of Figure 8, where the contrast is low, Prewitt and Sobel only extract a few edges. The Canny, Canny–Median, and WL operators can extract correct edges but miss some edge information. The CIED can extract complete edges, and the edges marked by the green rectangles can only be extracted by CIED. In the fifth row MI\_5 and sixth row MI\_6 of Figure 8, where the contrast is much weaker, Prewitt and Sobel almost extract nothing. Canny and Canny–Median only extract part of the edge information. The WL operator and the CIED extract better edges. As marked by the green rectangle, the CIED extracts clearer and more complete edges with less noise. From the visual results of edge detection, the visual performances of Prewitt and Sobel are similar; they can extract edge information, but many edges are lost at low contrast, and even no edge information is extracted at extreme contrast. Canny and Canny–Median visual performances are similar, with more false edges extracted at higher contrast, and incomplete edges extracted at lower contrast, but overall better than Prewitt and Sobel. The WL operator performs better than Canny and Canny–Median. It can still extract the edge information more completely in the case of weak contrast, but the WL algorithm is accompanied by more noise interference. CIED has the best visual performance, extracting complete contours despite very low contrast. In addition, the CIED has better continuity of edge line segments and less noise.

To further quantitatively analyze the edge detection methods, we also utilize accuracy, precision, recall, and F1-score to evaluate the performance of edge detection comprehensively. Table 2 shows the average metric results for different methods. The evaluation metric results of the CIED are indicated in italics, while the optimal values are highlighted in bold. The CIED method attained the highest average accuracy of 0.978, average precision of 0.408, average recall of 0.917, and average F1-score of 0.550 among all methods. These results indicate that CIED performs well. It shows that CIED can efficiently detect most target edges while balancing the corresponding accuracy. On the other hand, the CIED can

reduce edge omissions when dealing with low-contrast medical images. As the F1-score is an evaluation metric integrating both precision and recall, it gives a more balanced reflection of the performance of edge detection results. Figure 9 shows the result of the detected edge images using different methods in terms of F1-score. According to the result, CIED performs the best and most stably among these methods. Some methods can hardly detect any edges under low contrast, resulting in an F1-score close to zero, whereas CIED can still robustly extract effective edge information.

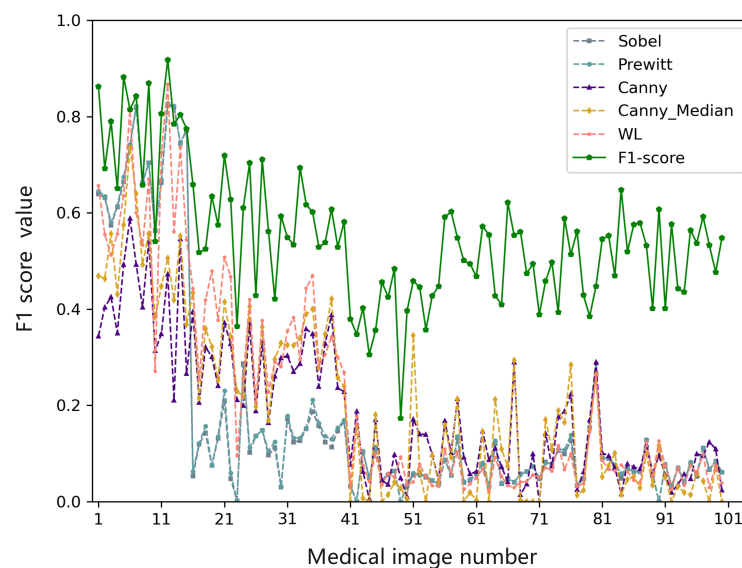


**Figure 8.** The demonstration of detected edge images using different methods. The green boxes highlight the edge detection results of CIED, while the red boxes indicate the results of other methods.

**Table 2.** The results of detected edge images using different methods in terms of accuracy, precision, recall, and F1-score. Data are presented as average values.

Methods	Accuracy	Precision	Recall	F1-Score
Prewitt	0.894	0.173	0.591	0.180
Sobel	0.894	0.171	0.582	0.176
Canny	0.967	0.163	0.266	0.190
Canny-Median	0.977	0.224	0.212	0.200
WL	0.813	0.158	0.806	0.218
CIED	<b><i>0.978</i></b>	<b><i>0.408</i></b>	<b><i>0.917</i></b>	<b><i>0.550</i></b>

The average evaluation metric results of the CIED are highlighted in italics, while the optimal average values of all methods are highlighted in bold.



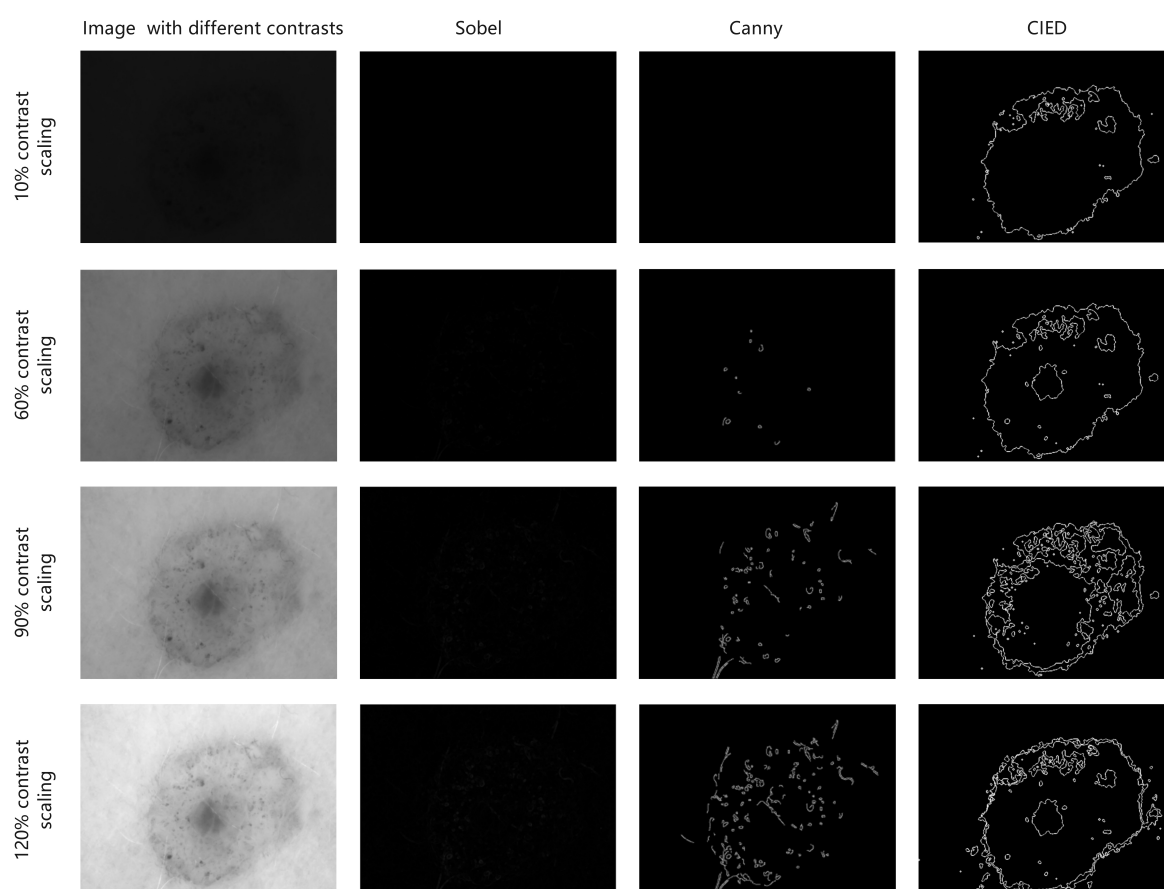
**Figure 9.** The results of detected edge images in terms of F1-score using different methods.

At low image contrast, the effect of the edges extracted by the edge detection methods is significantly weakened, and in extreme cases, it is not even possible to extract any effective edges. However, unlike other methods, the CIED can extract clear and effective edges under different image contrasts.

To more intuitively observe the performance of methods under different contrasts, we select an image, scale its original contrast in equal proportions, and then observe the edge images under different contrasts. Figure 10 shows the edge detection effect of Sobel, Canny, and the CIED under different contrasts on the same image. We demonstrate edge images for linear contrasts of 10, 60, 90, and 120%. In the case of the original image having low contrast, even if the contrast is increased to 120%, the visual effect of Sobel and Canny edge detection is still ineffective. The edge detection effect of the Sobel decreases with the decrease in the contrast, making it difficult to observe the edge information intuitively. It is not until the contrast is increased to 120% that the edge information is only visible in bits and pieces. The edge detection performance of the Canny also decreases with the decrease in contrast, and the edge information becomes clearer when the contrast is increased to 120%. However, the CIED can extract complete and clear edge information at all contrast levels, which demonstrates that the CIED is robust to contrast-invariant. To comprehensively compare the performance at different contrast levels, we record in detail the accuracy, precision, recall, and F1-score values of the three methods, Sobel, Canny, and the CIED, for contrasts ranging from 50% to 140%. Table 3 presents the specific evaluation metrics results for Sobel, Canny, and the CIED at different contrast levels, the outcomes of the CIED are indicated in italics, while the optimal values are highlighted in bold. From the result of Table 3, the precision and recall of Sobel and Canny gradually increase when the contrast is gradually enhanced, which indicates that there is an improvement in the edge detection ability when the contrast is enhanced. While the CIED is more stable in precision and recall in any contrast. Compared to Sobel and Canny, the precision, recall, and F1-score of the CIED are better than the other two methods and reach the highest precision of 0.246, recall of 0.793, and an F1-score of 0.371.

Some edge detection techniques based on gradients, such as Sobel and Canny, are not effective in low-contrast situations. The reason is that the grayscale difference between neighboring pixels in a low-contrast environment is extremely small. The Sobel is less sensitive to such small grayscale changes, and it is difficult to accurately locate the edge position. Then, the phenomenon of edge loss and edge blurring occurs. The Canny

primarily comprises Gaussian filtering, gradient magnitude computation, non-maximum value suppression, and double-threshold detection. In this method, the gradient magnitude computation is used to identify edges. The gradient magnitude computation step is also based on the grayscale difference between pixels. A weak contrast makes the gradient magnitude overall at a low level. With low contrast, weaker edges are easily suppressed due to the magnitude not reaching the threshold, resulting in incomplete edge detection. CIED shows significant advantages in cases of different contrasts, especially when the contrast is low. It extracts edge information by analyzing the three MSB planes and is almost independent of contrast variations since it does not depend on gray level differences. Not only that, but also when the contrast is reduced, it may also suppress disturbing factors such as noise to a certain extent. This process helps the CIED to recognize the edges better and thus achieve stable edge detection. It is indicated that CIED is robust to contrast-invariant.



**Figure 10.** The demonstration of detected edge images at different contrast levels using Sobel, Canny, and CIED methods.

**Table 3.** The comparison result of detected edge images at different contrast levels using Sobel, Canny, and the CIED.

Evaluation Metrics	50%	60%	70%	80%	90%	100%	110%	120%	130%	140%	Average
Accuracy—Sobel	0.975	0.965	0.954	0.940	0.924	0.905	0.882	0.858	0.829	0.809	0.905
Accuracy—Canny	0.990	0.989	0.988	0.986	0.983	0.981	0.979	0.976	0.974	0.972	<b>0.982</b>
Accuracy—CIED	0.975	0.984	0.972	0.969	0.964	0.975	0.956	0.970	0.966	0.963	<i>0.969</i>
Precision—Sobel	0.044	0.053	0.055	0.059	0.060	0.058	0.057	0.055	0.052	0.052	0.054
Precision—Canny	0.000	0.000	0.035	0.052	0.043	0.061	0.061	0.066	0.065	0.078	0.041
Precision—CIED	0.267	0.412	0.246	0.241	0.198	0.271	0.162	0.245	0.221	0.196	<b>0.246</b>
Recall—Sobel	0.079	0.154	0.232	0.332	0.438	0.533	0.642	0.713	0.782	0.835	0.438
Recall—Canny	0.000	0.000	0.011	0.025	0.034	0.064	0.083	0.112	0.128	0.175	0.061
Recall—CIED	0.768	0.767	0.809	0.778	0.794	0.802	0.767	0.834	0.798	0.809	<b>0.793</b>
F1-score—Sobel	0.056	0.079	0.089	0.100	0.105	0.105	0.105	0.102	0.097	0.098	0.093
F1-score—Canny	0.000	0.000	0.017	0.034	0.038	0.062	0.071	0.083	0.086	0.108	0.046
F1-score—CIED	0.397	0.536	0.377	0.368	0.318	0.406	0.268	0.378	0.346	0.316	<b>0.371</b>

The average results of the CIED are highlighted in italics, while the optimal values are highlighted in bold.

## 5. Discussion

In medical image-assisted diagnosis, accurate edge detection is crucial for clinical decisions such as diagnosis and treatment planning, which can help doctors quickly target the lesion area observe the location, size, and shape of the lesion, and then quickly carry out diagnosis and treatment. However, many of the existing edge detection methods rely on gradient computation and often face challenges in processing medical images, especially those with low contrast and blurred edges, which make it difficult to extract clear and complete edges [40]. To tackle these challenges, we put forward a novel edge detection method, CIED. The CIED preprocesses medical images by a combination of Gaussian filtering and morphological processing. After processing, the CIED utilizes three MSB planes, respectively. These planes are processed for neighborhood analysis and finally fused with the processed results to obtain the final edge detection results.

In the CIED method, it is necessary to use binary image information corresponding to the target image, which can usually be processed as known information. Therefore, we use the thresholding method to obtain the binary version of the target image, which is regarded as a preprocessing operation. The thresholding method used to binarize the target image can be regarded as a custom modular operation, and in this paper, we suggest choosing the Otsu thresholding method with better performance. In addition, we tested the experimental effects of image binarization pre-operation under different thresholding methods. We tested CIED based on different thresholding methods, which include Li [42], Sauvola [43], Yen [44], and Iteration [45]. Among them, the Li method determines the threshold by minimizing the cross-entropy between the foreground and the background. The Sauvola method is based on the local mean and standard deviation and introduces an adjustment parameter to determine the threshold. The Yen method determines the threshold by maximizing the entropy of the inter-class probability. The Iteration method obtains a stable threshold by performing multiple iterations based on the current average grayscale. We use these different thresholding methods to run the same group of random images and then calculate their evaluation metrics including accuracy, precision, recall, and F1-score, respectively. Table 4 shows the comparison results of the edge images detected by different automatic thresholding methods. Among them, the Otsu method performs the best. The mean indicators of Otsu all reach the highest. Its average accuracy rate is 0.971, the average precision rate is 0.505, the average recall rate is 0.925, and the average F1-score is 0.642. The Iteration is second. Its average accuracy rate is 0.970, the average

precision rate is 0.495, the average recall rate is 0.920, and the average F1-score is 0.632. The methods that follow closely are the Li and Sauvola methods, and their performances are similar, with the mean F1-scores reaching 0.566 and 0.577, respectively. The method with a relatively weaker performance is the Yen method, whose mean F1-score is 0.512. Finally, based on the experimental data, it can be clearly concluded that Otsu has the best overall result in the performance impact of different thresholding methods on the CIED method.

The CIED has more significant advantages over several traditional and improved edge detection methods, which include Prewitt, Sobel, and Canny as well as Canny–Median and WL operators. Prewitt and Sobel perform similarly in that is difficult to accurately locate edge positions with low contrast due to limited sensitivity to small changes in pixel intensity, resulting in weak edge strength and visually almost invisible clear edges [11]. Canny and Canny–Median algorithms perform similarly. To further eliminate the noise in the image, Canny–Median adds the processing of median filtering to the image, which further suppresses the image noise [20]. Canny and Canny–Median are equally susceptible to their gradient magnitude calculations at low contrasts, resulting in weaker edges being suppressed because the magnitude does not reach the threshold, thus making edge detection incomplete, while at higher contrasts they extract more false edges, and the performance of edge detection is not stable at different contrasts. The WL algorithm calculates the local intensity mean of the  $3 \times 3$  neighborhood and the local signal energy variation between the neighborhoods, and it determines whether it is an edge point based on the calculated value of the local signal energy variation. It makes the results of the edge detection highly adaptive, and it has good directional symmetry [22]. Compared to the previous methods, WL extracts more complete edge information in images with low contrast, although it also extracts more noise at the same time. The CIED performs best overall, especially in extracting complete and clear edge information even in the case of low-contrast medical images. The CIED method does not rely on the grayscale calculation between pixels; it utilizes three MSB plane information under the binary representation of the grayscale value and is able to stably extract the edge information in different contrast cases, especially in low-contrast cases, independent of the contrast changes. Through both intuitive and quantitative evaluation, the CIED has good performance. Qualitative analysis shows that CIED is able to extract more complete and continuous edge line segments with less noise. The quantitative analysis further highlights the significant advantage of the CIED. All three metrics achieved their highest values, with a mean precision of 0.408, a mean recall of 0.917, and a mean F1-score of 0.550. It indicates that our algorithm can have better edge detection performance in different types of medical images. To more intuitively observe the performance of different methods at different contrasts, we scaled the original contrast of the medical images in equal proportions and detected the edges. The CIED extracts complete and clear contour information at any contrast which indicates that the CIED has a high degree of robustness at different contrasts.

**Table 4.** The comparison result of detected edge images at different auto threshold methods.

Evaluation Metrics	IMG_01	IMG_02	IMG_03	IMG_04	IMG_05	IMG_06	IMG_07	IMG_08	IMG_09	IMG_10	Average
Accuracy—Li	0.926	0.928	0.931	0.965	0.961	0.957	0.990	0.985	0.989	0.991	0.962
Accuracy—Sauvola	0.897	0.910	0.910	0.960	0.957	0.953	0.993	0.989	0.993	0.995	0.955
Accuracy—Yen	0.897	0.879	0.888	0.964	0.965	0.961	0.988	0.973	0.987	0.991	0.949
Accuracy—Iteration	0.939	0.944	0.945	0.965	0.981	0.969	0.993	0.989	0.993	0.991	0.970
Accuracy—Otsu	0.939	0.944	0.945	0.965	0.981	0.972	0.993	0.988	0.993	0.991	<b>0.971</b>
Precision—Li	0.501	0.602	0.735	0.404	0.339	0.337	0.287	0.422	0.271	0.341	0.423
Precision—Sauvola	0.410	0.543	0.661	0.371	0.360	0.324	0.350	0.422	0.382	0.474	0.429
Precision—Yen	0.401	0.444	0.579	0.405	0.412	0.396	0.217	0.240	0.226	0.380	0.370
Precision—Iteration	0.544	0.667	0.836	0.408	0.565	0.365	0.350	0.435	0.444	0.337	0.495
Precision—Otsu	0.544	0.667	0.836	0.409	0.565	0.473	0.350	0.425	0.444	0.337	<b>0.505</b>
Recall—Li	0.944	0.960	0.978	0.885	0.841	0.784	0.745	0.979	0.835	0.974	0.892
Recall—Sauvola	0.952	0.977	0.974	0.911	0.985	0.830	0.613	0.953	0.835	0.974	0.900
Recall—Yen	0.937	0.962	0.956	0.950	0.984	0.910	0.719	0.966	0.835	0.982	0.920
Recall—Iteration	0.951	0.970	0.933	0.961	0.987	0.929	0.613	0.953	0.934	0.974	0.920
Recall—Otsu	0.951	0.970	0.933	0.962	0.987	0.961	0.613	0.971	0.934	0.974	<b>0.925</b>
F1-score—Li	0.655	0.740	0.839	0.555	0.484	0.472	0.414	0.590	0.409	0.506	0.566
F1-score—Sauvola	0.574	0.698	0.787	0.527	0.528	0.466	0.445	0.585	0.524	0.638	0.577
F1-score—Yen	0.562	0.608	0.721	0.568	0.581	0.552	0.334	0.385	0.356	0.548	0.512
F1-score—Iteration	0.692	0.790	0.882	0.573	0.719	0.524	0.445	0.597	0.602	0.500	0.632
F1-score—Otsu	0.692	0.790	0.882	0.574	0.719	0.634	0.445	0.591	0.602	0.500	<b>0.642</b>

The optimal average evaluation metrics results are highlighted in bold.

The CIED method adopts the combination of Gaussian filtering and morphological processing to preprocess medical images. In the morphological processing, the processing method of opening operation followed by closing operation is adopted. Gaussian filtering can effectively remove image noise and better preserve edge details, thus improving the image quality. In morphological processing, the opening operation can remove small objects, burrs, and separate connected objects, making the main part of the image clearer and facilitating the accurate identification of object edges. Meanwhile, the closing operation can fill the internal holes of the target object, making the edges of the object more complete and continuous. The preprocessing method combining Gaussian filtering and morphological processing creates favorable conditions for subsequent edge detection and plays an important role in improving the accuracy and effect of CIED edge detection. The CIED method is based on three MSB planes. It can highlight important features, reduce the effect of noise, increase contrast, and improve computational efficiency [46]. Three MSB planes contain the main structural and contour information within the image, and these significant shapes and structural details are emphasized. Usually, the noise is more obvious in the LSB planes, while the information in the three MSB planes is more stable and reliable [47]. In the process of edge detection, choosing the three MSB planes can reduce the interference of noise in edge detection results and improve the performance of edge detection. The three MSB planes enhance the contrast of an image without additional contrast processing, making edges sharper. By highlighting different gray-level regions in the image, the three MSB planes enhance edge detection by increasing the contrast between the edges and the background. This is especially useful for some images with low contrast. Utilizing three MSB planes for edge detection also improves computational efficiency. This is because the CIED is based on bit planes. It does not use complex pixel values for calculation. Instead, it uses simple binary values of zero and one, so it is faster and more efficient. Based on the computational characteristics of CIED, in order to further



improve the real-time performance of CIED, we explore the application of the CIED method to edge devices or embedded systems to achieve real-time edge detection. These devices usually have limited computing resources, and further algorithm optimization is required to ensure real-time performance. We are considering using frameworks such as CUDA or TensorRT to optimize the computational efficiency of CIED in future research so that it can run on embedded systems. This can help expand the application of CIED in fields such as real-time medical image analysis, and assist with creating patient-centered applications [48].

In future research, to enhance the adaptability and accuracy of the CIED, we plan to progress step by step from ensemble learning methods, through machine learning methods to deep learning methods. First, through ensemble learning, based on the CIED method, multiple edge detection operators can be integrated. Through ensemble strategies such as voting and weighted averaging, we will try to enhance the robustness and detection precision. Second, within the bit plane, various image features such as gradient information and texture information can be extracted. A multi-feature fusion strategy will be designed. Via feature selection and feature fusion techniques, the richness and representativeness of the features will be improved. Traditional machine learning models such as Support Vector Machine (SVM) and Random Forest (RF) will be introduced to classify and optimize the extracted bit plane features, further enhancing the detection effect. Finally, we will explore deep learning methods, employing models such as Convolutional Neural Network (CNN) and U-Net to automatically learn edge features from bit planes, and try to further improve the performance of CIED.

## 6. Conclusions

In this study, we propose the CIED method to address the challenges faced by existing gradient-based methods in processing low-contrast medical images with blurred edges. The method selects the three MSB planes to extract edge information from medical images. This is due to the three MSB planes containing important contour features. Using bit planes for edge detection reduces the effect of noise, naturally enhances contrast, and improves computational efficiency. The CIED is thoroughly compared with existing edge detection methods. Qualitative analysis shows that the edges extracted by the CIED are more complete, continuous, and less noisy. Quantitative analysis shows that the CIED demonstrates superior performance with a mean precision of 0.408, a mean recall of 0.917, and a mean F1-score of 0.550. It has good robustness under different contrasts, and the CIED can extract complete and valid edges even at very low contrasts. The CIED provides a new approach in the domain of medical image-assisted diagnosis, which contributes to enhancing the performance of disease diagnosis and the effectiveness of treatment. In future research, we will try to explore and extend the CIED method to non-medical image applications such as industrial defect detection, satellite image analysis, natural scene image processing, and remote sensing image processing. These extensions can further validate the universality and robustness of the CIED method and enhance its practical value in different fields.

**Author Contributions:** D.L.: Conceptualization, Methodology, Data Curation, Software, Visualization, Writing—Original Draft. P.C.-I.P.: Conceptualization, Supervision, Validation, Writing—Review and Editing. C.-K.L.: Methodology, Writing—Review and Editing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Macao Science and Technology Development Fund (funding ID: 0088/2023/ITP2) and Macao Polytechnic University research grant (funding ID: RP/FCA-13/2022; submission code: fca.e22a.7fee.c).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The dataset is publicly available at [www.kaggle.com](https://www.kaggle.com) (accessed on 12 January 2025).

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

## References

1. Jing, J.; Liu, S.; Wang, G.; Zhang, W.; Sun, C. Recent advances on image edge detection: A comprehensive review. *Neurocomputing* **2022**, *503*, 259–271.
2. Tajbakhsh, N.; Shin, J.Y.; Gurudu, S.R.; Hurst, R.T.; Kendall, C.B.; Gotway, M.B.; Liang, J. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Trans. Med. Imaging* **2016**, *35*, 1299–1312.
3. Zhou, M.; Li, C.; Cheng, M.; Zhao, S. Medical thermal effects and CT radiation image examination in the treatment of sequelae of chronic pelvic inflammatory disease with warm acupuncture therapy: A meta-analysis. *Therm. Sci. Eng. Prog.* **2024**, *57*, 103149.
4. Yang, D.; Peng, B.; Al-Huda, Z.; Malik, A.; Zhai, D. An overview of edge and object contour detection. *Neurocomputing* **2022**, *488*, 470–493.
5. Drazkowska, M. Detection of Pediatric Femur Configuration on X-ray Images. *Appl. Sci.* **2021**, *11*, 9538.
6. Dołęga-Dołęgowski, D.; Dolega-Dolegowska, M.; Pregowska, A.; Malinowski, K.; Proniewska, K. The application of mixed reality in root canal treatment. *Appl. Sci.* **2023**, *13*, 4078.
7. Rajpurkar, P.; Lungren, M.P. The current and future state of AI interpretation of medical images. *N. Engl. J. Med.* **2023**, *388*, 1981–1990.
8. Orujov, F.; Maskeliūnas, R.; Damaševičius, R.; Wei, W. Fuzzy based image edge detection algorithm for blood vessel detection in retinal images. *Appl. Soft Comput.* **2020**, *94*, 106452.
9. Nelke, K.; Janeczek, M.; Pasicka, E.; Żak, K.; Barnaś, S.; Nienartowicz, J.; Gogolewski, G.; Maag, I.; Dobrzyński, M. The Occurrence of a Rare Mandibular Retromolar Triangle Schwannoma and Its Differentiation from Other Rare and Atypical Oral Cavity Tumours. *Appl. Sci.* **2024**, *14*, 3924.
10. Gaheen, M.A.; Ibrahim, E.; Ewees, A.A. Edge detection-based segmentation for detecting skin lesions. In *Machine Learning, Big Data, and IoT for Medical Informatics*; Elsevier: Berlin/Heidelberg, Germany, 2021; pp. 127–142.
11. Shrivakshan, G.; Chandrasekar, C. A comparison of various edge detection techniques used in image processing. *Int. J. Comput. Sci. Issues (IJCSI)* **2012**, *9*, 269.
12. Sobel, I.E. *Camera Models and Machine Perception*; Stanford University: Stanford, CA, USA, 1970.
13. Zhang, W.; Zhao, Y.; Breckon, T.P.; Chen, L. Noise robust image edge detection based upon the automatic anisotropic Gaussian kernels. *Pattern Recognit.* **2017**, *63*, 193–205.
14. Li, Z.; Shu, H.; Zheng, C. Multi-scale single image dehazing using Laplacian and Gaussian pyramids. *IEEE Trans. Image Process.* **2021**, *30*, 9270–9279.
15. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, *6*, 679–698.
16. Rajan, R.; Kumar, S. Gauss gradient algorithm for edge detection in retinal optical coherence tomography images. *Procedia Comput. Sci.* **2023**, *218*, 1014–1026.
17. Mittal, M.; Verma, A.; Kaur, I.; Kaur, B.; Sharma, M.; Goyal, L.M.; Roy, S.; Kim, T.H. An efficient edge detection approach to provide better edge connectivity for image analysis. *IEEE Access* **2019**, *7*, 33240–33255.
18. Hien, N.M.; Binh, N.T.; Viet, N.Q. Edge detection based on Fuzzy C Means in medical image processing system. In Proceedings of the 2017 International Conference on System Science and Engineering (ICSSE), Ho Chi Minh City, Vietnam, 21–23 July 2017; pp. 12–15.
19. Nikolic, M.; Tuba, E.; Tuba, M. Edge detection in medical ultrasound images using adjusted Canny edge detection algorithm. In Proceedings of the 2016 24th Telecommunications Forum (TELFOR), Belgrade, Serbia, 22–23 November 2016; pp. 1–4.
20. Topno, P.; Murmu, G. An improved edge detection method based on median filter. In Proceedings of the 2019 Devices for Integrated Circuit (DevIC), Kalyani, India, 23–24 March 2019; pp. 378–381.
21. Elmi, S.; Elmi, Z. A robust edge detection technique based on Matching Pursuit algorithm for natural and medical images. *Biomed. Eng. Adv.* **2022**, *4*, 100052.
22. Lin, W.C.; Wang, J.W. Edge detection in medical images with quasi high-pass filter based on local statistics. *Biomed. Signal Process. Control.* **2018**, *39*, 294–302.
23. Rahman, S.; Uddin, J.; Khan, H.U.; Hussain, H.; Khan, A.A.; Zakarya, M. A novel steganography technique for digital images using the least significant bit substitution method. *IEEE Access* **2022**, *10*, 124053–124075.

24. Chen, F.; Yuan, Y.; He, H.; Tian, M.; Tai, H.M. Multi-MSB compression based reversible data hiding scheme in encrypted images. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 905–916.
25. Setiadi, D.R.I.M. Improved payload capacity in LSB image steganography uses dilated hybrid edge detection. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 104–114.
26. Abhisheka, B.; Biswas, S.K.; Purkayastha, B. A comprehensive review on breast cancer detection, classification and segmentation using deep learning. *Arch. Comput. Methods Eng.* **2023**, *30*, 5023–5052.
27. Ye, Z.; Zhao, R.; Zhang, Y.; Xiao, X.; Lan, R.; Xiang, Y. Noise-free thumbnail-preserving image encryption based on MSB prediction. *Inf. Sci.* **2022**, *617*, 395–415.
28. Jebur, S.A.; Nawar, A.K.; Kadhim, L.E.; Jahefer, M.M. Hiding Information in Digital Images Using LSB Steganography Technique. *Int. J. Interact. Mob. Technol.* **2023**, *17*, 167–178.
29. Xia, X.; Zhang, S.; Wang, K.; Gao, T. A novel color image tampering detection and self-recovery based on fragile watermarking. *J. Inf. Secur. Appl.* **2023**, *78*, 103619.
30. Dubey, S.R.; Singh, S.K.; Singh, R.K. Local bit-plane decoded pattern: a novel feature descriptor for biomedical image retrieval. *IEEE J. Biomed. Health Inform.* **2015**, *20*, 1139–1147.
31. Yao, Y.; Wang, K.; Chang, Q.; Weng, S. Reversible data hiding in encrypted images using global compression of zero-valued high bit-planes and block rearrangement. *IEEE Trans. Multimed.* **2023**, *26*, 3701–3714.
32. Taneja, N.; Mishra, G.S.; Bhardwaj, D. A Bit-Plane Slicing Technique for the Classification of Anti-forensically Contrast-Enhanced Images. In Proceedings of the 2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT), Greater Noida, India, 9–10 February 2024; Volume 5, pp. 1619–1623.
33. Wen, H.; Lin, Y.; Kang, S.; Zhang, X.; Zou, K. Secure image encryption algorithm using chaos-based block permutation and weighted bit planes chain diffusion. *IScience* **2024**, *27*, 108610.
34. Faheem, Z.B.; Hanif, D.; Arslan, F.; Ali, M.; Hussain, A.; Ali, J.; Baz, A. An edge inspired image watermarking approach using compass edge detector and LSB in cybersecurity. *Comput. Electr. Eng.* **2023**, *111*, 108979.
35. Zhai, G.; Liang, Y.; Tan, Z.; Wang, S. Development of an iterative Otsu method for vision-based structural displacement measurement under low-light conditions. *Measurement* **2024**, *226*, 114182.
36. Anjum, N.; Sakib, A.N.M.; Masudul Ahsan, S.M. Classification of brain hemorrhage using deep learning from CT scan images. In Proceedings of the International Conference on Information and Communication Technology for Development: ICICTD 2022, Khulna, Bangladesh, 29–30 July 2022; Springer: Singapore, 2023; pp. 181–193.
37. Tschandl, P.; Rosendahl, C.; Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Sci. Data* **2018**, *5*, 1–9.
38. Yacouby, R.; Axman, D. Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models. In Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems, Online, 20 November 2020; pp. 79–91.
39. Tariq, N.; Hamzah, R.A.; Ng, T.F.; Wang, S.L.; Ibrahim, H. Quality assessment methods to evaluate the performance of edge detection algorithms for digital image: A systematic literature review. *IEEE Access* **2021**, *9*, 87763–87776.
40. Amer, G.M.H.; Abushaala, A.M. Edge detection methods. In Proceedings of the 2015 2nd World Symposium on Web Applications and Networking (WSWAN), Sousse, Tunisia, 21–23 March 2015; pp. 1–7.
41. El-Khamy, S.E.; Lofly, M.; El-Yamany, N. A modified fuzzy Sobel edge detector. In Proceedings of the Seventeenth National Radio Science Conference, 17th NRSC'2000 (IEEE Cat. No. 00EX396), Minufiya, Egypt, 24 February 2000; pp. C32–1.
42. Sezgin, M.; Sankur, B.I. Survey over image thresholding techniques and quantitative performance evaluation. *J. Electron. Imaging* **2004**, *13*, 146–168.
43. Senthilkumaran, N.; Vaithegi, S. Image segmentation by using thresholding techniques for medical images. *Comput. Sci. Eng. Int. J.* **2016**, *6*, 1–13.
44. Bunyaviorch, L.; Thanyawet, N. Comparison of Thresholding Methods in Image Processing for Uterus Ultrasound Images. In Proceedings of the RSU International Research Conference 2021 on Science and Technology, Riga, Latvia, 30 April 2021; pp. 469–476.
45. Huang, M.; Liu, Y.; Yang, Y. Edge detection of ore and rock on the surface of explosion pile based on improved Canny operator. *Alex. Eng. J.* **2022**, *61*, 10769–10777.
46. Verma, A.K.; Sarkar, T. Utilizing Imaging Steganographic Improvement using LSB & Image Decoder. In Proceedings of the 2024 International Conference on Communication, Computer Sciences and Engineering (IC3SE), Gautam Buddha Nagar, India, 9–11 May 2024; pp. 144–150.

47. Singh, R.; Patel, S.; Vaish, A. Secure most significant bit plane compression based reversible data hiding in encrypted image technique using Huffman Ciphersystem. *J. Electron. Imaging* **2024**, *33*, 013020.
48. Pang, P.C.-I.; Chang, S.; Verspoor, K.; Pearce, J. Designing Health Websites Based on Users' Web-Based Information-Seeking Behaviors: A Mixed-Method Observational Study. *J. Med. Internet Res.* **2016**, *18*, e145.

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