



## A Simple and Effective Edge Detection Algorithm Based on Boolean Logic

Mohammed Ali Tawfeeq<sup>1</sup> and Mahmood Zeki Abdallah<sup>2</sup>

### ABSTRACT

Detecting the edges of objects in images is an essential issue in the field of computer vision and image processing, especially in light of the increasing need for immediate and online interaction in determining the content of these images, which requires adopting an appropriate algorithm. This paper introduced a new Simple and Effective Edge Detection (SEED) algorithm. This algorithm relies on Boolean operations to detect edges in binary digital images. SEED analyses every pair of adjacent pixels, horizontally and vertically, in a smoothly and easily manner. This algorithm showed high performance in identifying edges with advanced ability to overcome false edges. To evaluate the SEED algorithm, it has been compared with both the Sobel and Canny algorithms by adopting quantitative evaluation metrics such as the peak signal to noise ratio (PSNR) and the mean square error (MSE), in addition to the intersection-to-union (IoU) ratio index, or what is called Jaccard. The values of the above metrics reflected a higher performance of the proposed algorithm. It has been also found that the detection rate of false edges decreased significantly, making it an effective tool for applications in this field.

### Article information:

**Keywords:** Binary Images, Boolean Logic, Canny, Sobel, Jaccard Index

### Article history:

Received: February 13, 2024

Revised: March 28, 2024

Accepted: May 2, 2024

Published: May 25, 2024

(Online)

DOI: [10.37936/ecti-cit.2024183.255695](https://doi.org/10.37936/ecti-cit.2024183.255695)

### 1. INTRODUCTION

Edge detection is an important and sensitive issue in image processing and computer vision applications, where this process is usually performed to identify and diagnose objects in images. The location where a significant change in brightness or color occurs in an image is often represented by an edge [1]. Identifying edges leads to distinguishing and identifying objects or features in images [2]. Detecting and identifying edges results in binary images, which paves the way for analyzing the content of the images, segmenting them, and identifying the objects in their contents [3]. Several edge detection techniques have been commonly used, such as the Sobel, Prewitt [4], Roberts [5], and Canny [6]. Although there are many mathematical methods that estimate the gradient of the image intensity function, all of these algorithms share a basic goal: identifying regions within the image where there is a noticeable shift in brightness [7, 8].

Many edge detection techniques based on machine learning have been developed [9-13].

By identifying the edges of the image, it is possible to recognize the topology of shapes and determine their features [14]. Still, the possibility that these edges contain some of adjacent pixels may increase the width of the edge and negatively affect the recognition of these features, which often requires the use of some algorithms to thin these edges [15, 16].

Li X. *et al.* (2020) [17] introduced an edge detection algorithm for images representing cancer cases using deep learning. The first stage included creating a 3-D model to reconstruct the surface structure of the cancer image. Next, an edge contour feature extraction method had been used to extract the complex features of cancer cells within the image. Then, a recombination model established for the multi-dimensional pixel feature distribution of the cancer image, using an accurate feature segmentation method to accomplish regional fusion and information recombination. Chen J. *et al.* (2021) [18], presented a novel approach for edge detection using cluster spectral curvature. The used strategy reformulates the challenge of edge detection as a classification problem, taking advantage of the distinct spatial

<sup>1,2</sup> The authors are with the Computer Engineering Department, College of Engineering, Mustansiriyah University, Baghdad, Iraq. E-mail: [drmatawfeeq@uomustansiriyah.edu.iq](mailto:drmatawfeeq@uomustansiriyah.edu.iq) and [drmzaali@uomustansiriyah.edu.iq](mailto:drmzaali@uomustansiriyah.edu.iq)

properties of image edge points, homogeneous points, and noise points. The proposed method efficiently separates homogeneous points from edge points. Liu and Zhu (2021) [19] utilized a subpixel edge detection algorithm to fulfill the stringent accuracy needs for locating spot edges in beam quality detection. The first stage involved using a Sobel operator to approximately locate the point of light and then repositioning the edge of the initially acquired pixel plane. Finally, identify the true sub-pixel edge points in the image based on the improved edge judgment criteria, thus achieving the sub-pixel edge extraction of the light spot image. Ultimately, they identified the genuine subpixel edge points in the image based on enhanced edge judgment criteria, thereby accomplishing the subpixel edge extraction of the light spot image. Sun H. *et al.* (2022) [20] put forward an improved flame edge detection method that uses a convolutional neural network (CNN). This approach is practical for creating and extracting edge maps. The grid structure used for this purpose mostly based on VGG16. Wu Ji *et al.* (2022) [21] presented a deep learning-based approach for edge detection in single-mode multimode fiber imaging. This approach uses a neural network that accepts a one-dimensional light intensity sequence as its input and produces the edge detection results of the target as its output. This neural network can recognize edge information of multiple, non-specific forms. The results showed that the proposed technique can determine edge details even if the sampling rate is low. Yan J *et al.* (2022) [22] proposed a model called dynamic threshold-directed neural P systems, or ODTNP systems for short. These systems combine the features of dynamic threshold and elevation mechanisms, considering both the direction and magnitude of the gradient. The results demonstrated the ability of this method to detect edges efficiently. A similar computational model of neurobehavior is spiking neural nonlinearity system (NSNP). Sheyan R. *et al.* (2023) [23], modified this system for custom image edge detection. The system runs on two convolutional cores supported by Particle Swarm Optimization (PSO) technology. Test results of this system gave effective results on standard images.

The ability of traditional algorithms and some machine learning techniques to effectively identify and map object edges within images has made these methods widely used. Some of them have proven to be highly efficient, but they also have drawbacks and complexities, as some methods involve using several algorithms to achieve the result, and this requires many calculations, which may increase the computational burden and limit the practical use of real time applications. Some of these algorithms also require pre-training processes that need to prepare many models to cover the state-space. Moreover, reaching generalization requires several epochs, which

takes a fair amount of time during this stage. On the other hand, some of these algorithms required carefully selecting thresholds and specifying parameters that can be adjusted, and their results may be affected by noise, in addition to the high detection rate of false edges in some of them.

This paper presents a new simple and effective edge detection (SEED) algorithm in binary images, based on Boolean logical operations. The computational ability and efficiency of Boolean logic in binary systems, in addition to its simplicity, made it an effective tool for comparing pixel intensities and deciding whether to consider it an edge or not. The proposed algorithm has been compared with well-known standard algorithms such as Sobel and Canny. The comparison results showed the outstanding performance of the proposed SEED algorithm with high PSNR and low MSE values, in addition to achieving a higher value of the Jaccard index with shallow false edges. On the other hand, the edges of the image extracted using this algorithm do not require any thinning. The results achieved using the proposed technique represent a significant contribution to the field of image processing. The results achieved using the proposed technique represent a significant contribution to the field of image processing, which can be reflected in many practical applications.

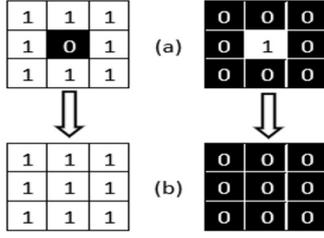
The remainder the paper follows this organization: Section 2 provides a detailed explanation of the SEED algorithm. In Section 3 the results of the algorithm, tested against a set of images, are presented, along with a comparative analysis. Finally, Section 4 presents the conclusions drawn.

## 2. PROPOSED EDGE DETECTION ALGORITHM

Boolean logic is characterized by its simple method and computational efficiency in dealing with binary data, which made it an incentive to rely on it in designing an algorithm to detect the edges of objects in binary images. This requires converting color and grayscale images into binary images. The basic principle on which the algorithm works is to process each successive pair of pixels horizontally and vertically to determine whether there is an edge or not based on a model designed for this purpose.

The SEED algorithm operates in two distinct phases. The first phase is intended to adjust the value of any isolated pixel in a 3x3 window that is distinct from its surroundings, as this distinction could suggest that the pixel represents noise, and might be processed.

Specifically, if the central pixel value of a window differs from all the pixel values surrounding it, it will be considered noise. The center pixel's value is then adjusted to match that of its surrounding pixels, as shown in Fig. 1. In other words, the noise removal process in this phase works exclusively on isolated



**Fig.1:** (a) Isolated Noisy Pixel, (b) Noise Removal.

pixels caused by salt and pepper noise, also known as impulsive noise, which presents itself as sparsely black and white pixels [24]. In this phase, to ensure uniform treatment of all pixels in the image of size  $(M*N)$ , a temporary row with zero values will be padded before the first row and after the last row. Similarly, a temporary column of zero values will be padded on the left and the right of the image.

To remove the noise from the pixels depicted on the left part of Fig. 1(a), the Boolean operation of Eq. (1) is adopted for any arbitrary pixel  $I(i,j)$  in the image, for  $i = (2, 3, \dots, M-1)$ ,  $j = (2, 3, \dots, N-1)$ ,  $k = (-1, 0, 1)$ , and  $l = (-1, 0, 1)$ :

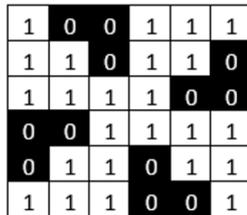
$$I(i,j) = I(i,j) \cap [I(i+k,j) \cup I(i,j+l)] \quad (1)$$

And, for the pixels depicted on the right part of Fig. 1(a), the corresponding bitwise relation used to remove the noise is:

$$I(i,j) = I(i,j) \cup [I(i+k,j) \cap I(i,j+l)] \quad (2)$$

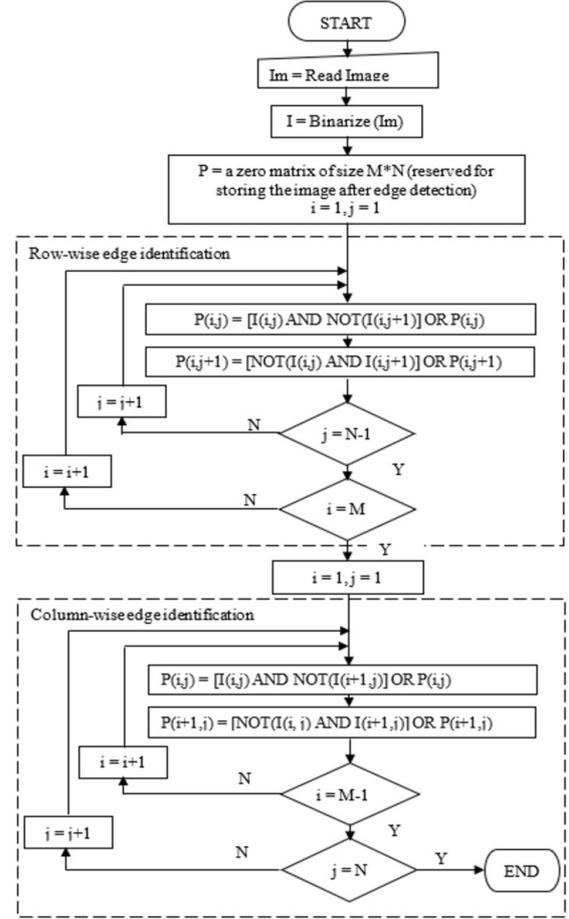
for  $(i+k, j+l) \neq (i,j)$ .

After removing noise from all rows and columns, as directed by Eqs. (1) and (2), the two padded rows and the two padded columns will be deleted, to return the image to its original size. The second stage task depends on identifying the edges of the image using the model shown in Figure 2. This model came from collecting all the possibilities that a pixel could be in an edge position.



**Fig.2:** Proposed Edge Prototype Model.

In the model provided in Fig. 2, white pixels symbolize data and are assigned a value of 1 in the binarized image, otherwise the pixel value is zero. Figure 3 shows the flowchart that uses Boolean algebraic relationships based on the model of Figure 2. The



**Fig.3:** Edge detection Flowchart in SEED algorithm.

model identifies edges based on the relative values of successive pixels, at the row level or at the column level. Specifically, if a pixel  $P$  with a value of one is immediately followed by a pixel with a value of zero, or vice versa, the pixel with value one is identified as an edge. In all other circumstances, the pixel is not considered an edge. In row-wise, the adopted Boolean logic operations are:

$$P(i,j) = [I(i,j) \cap NOT(I(i,j+1))] \cup P(i,j) \quad (3)$$

$$P(i,j+1) = [NOT(I(i,j)) \cap I(i,j+1)] \cup P(i,j+1) \quad (4)$$

Where “ $\cap$ ” and “ $\cup$ ” represent the “AND” and the “OR” operations, respectively. Likewise for columns, the logical operations for edge detection are:

$$P(i,j) = [I(i,j) \cap NOT(I(i+1,j))] \cup P(i,j) \quad (5)$$

$$P(i+1,j) = [NOT(I(i,j)) \cap I(i+1,j)] \cup P(i+1,j) \quad (6)$$

### 3. RESULTS

This work tested several images using the proposed system, adopting MATLAB R2019b 64-bit (Win64).

The tested images represent two sets of images related to different applications. The first set of images were clear and free of noise, while the second set included the same images with a certain level of noise added before the edge detection process.

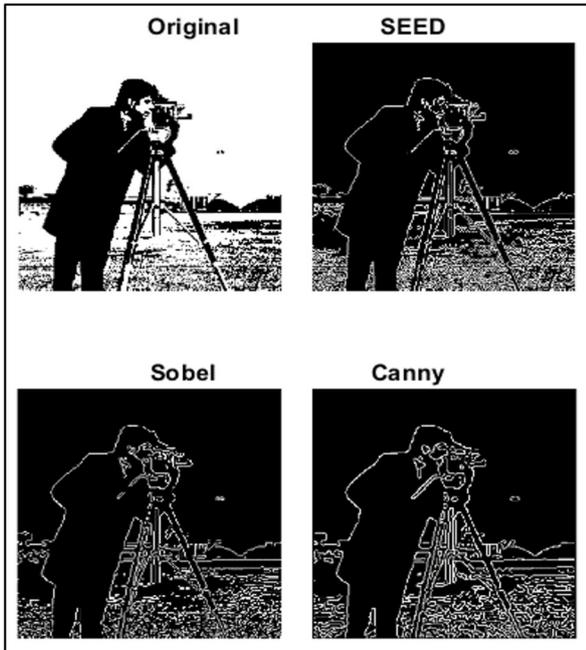


Fig.4: Camera Man Edge Detection.

The resulting edge detection outcomes compared to those produced by two well-known standard edge detection algorithms, Sobel and Canny. Noise-free images do not need to apply the first stage of the SEED algorithm. Thus, only the second phase, which is responsible for edge detection, was used. Fig. 4 to Fig. 7 show the results from the first set of images. The top-left images in these figures represent the bitmap versions of the original images.

For the purpose of verifying the results and evaluating the presence of false edges, a certain segment of Figure 4 has been enlarged, as shown in Figure 8. It is evident that the proposed algorithm produces free of false edges results, in contrast to those made by the Sobel and Canny methods. Additionally, the edges detected using the SEED algorithm show no need for thinning.

The same applies to Figures 5-7, where specific locations were selected and Magnified, as shown in Figures 9-11.

Performance evaluation also involved calculating the peak signal-to-noise ratio (PSNR) and the mean square error (MSE) as presented in Table 1, where the SEED algorithm recorded the maximum PSNR and the minimum MSE for all tested images.

The Jaccard Index, also called Intersection over

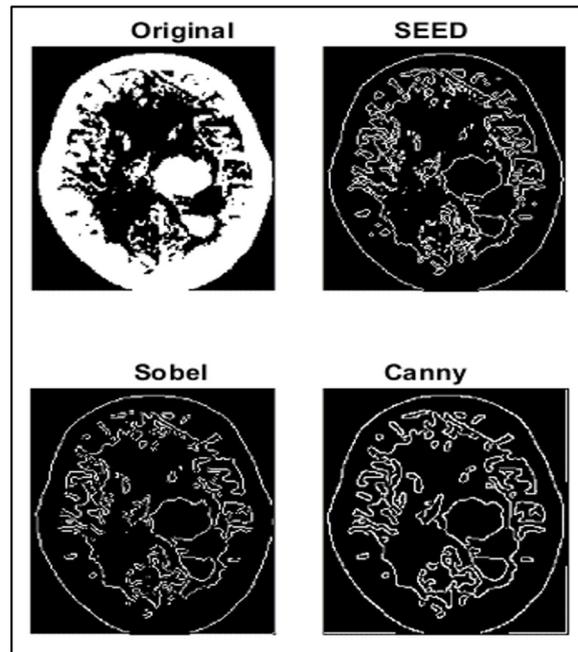


Fig.5: Brain CT-Scan Edge Detection.

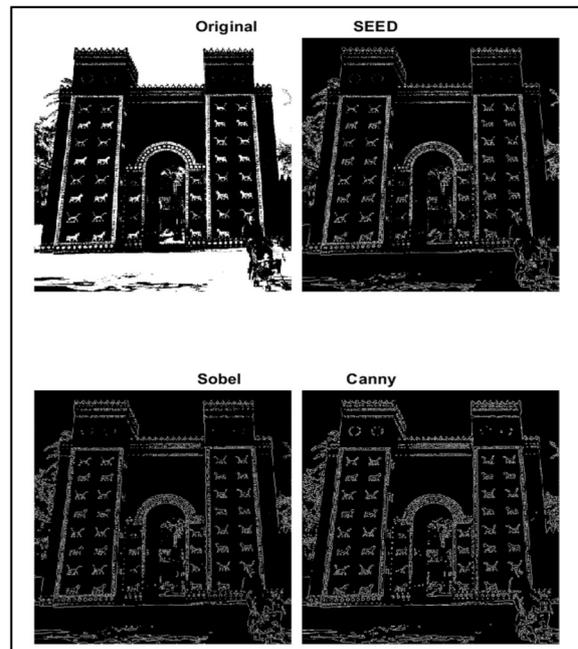


Fig.6: Ishtar-Gate Edge Detection.

Table 1: PSNR and MSE for Noise-Free Images.

Image	PSNR			MSE		
	SEED	Sobel	Canny	SEED	Sobel	Canny
Camera Man	2.71	1.93	1.98	0.53	0.64	0.63
CT-Scan	4.61	3.55	3.39	0.34	0.44	0.45
Ishtar Gate	5.2	4.24	4.07	0.30	0.37	0.39
Flowers	4.96	4.56	4.51	0.31	0.34	0.35

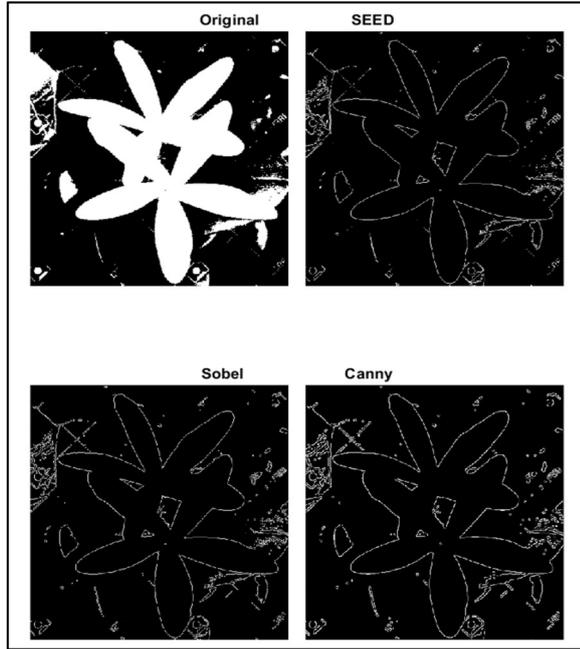


Fig.7: Flowers Image Edge Detection.

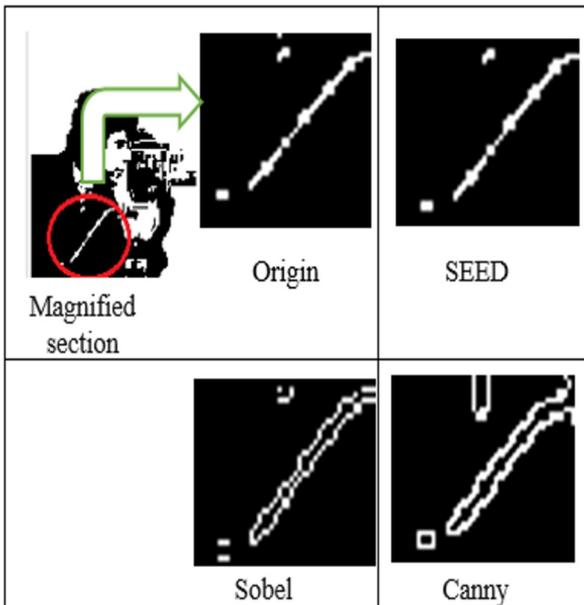


Fig.8: Searching for False Edges in 'Camera Man'.

Union (IoU), measures the similarity between two binary masks or two binary images. IoU scores span from 0 (indicating no overlap) to 1 (indicating complete overlap). It offers a straightforward numerical representation of the alignment between detected edges and their corresponding ground truth. A higher IoU indicates a closer match. It's worth noting that the thickness of the detected edge compared to the ground truth can influence the IoU score. Table 2 shows the calculated Jaccard index for the tested noise-free images.

Table 2 clearly shows that the Jaccard index results for the SEED algorithm are superior to those of

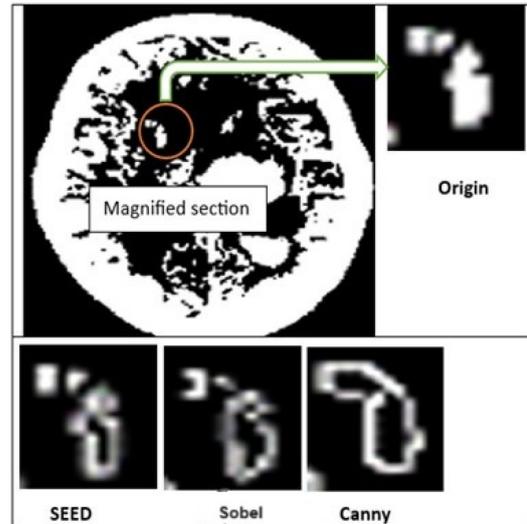


Fig.9: Searching for False Edges in 'Brain'.

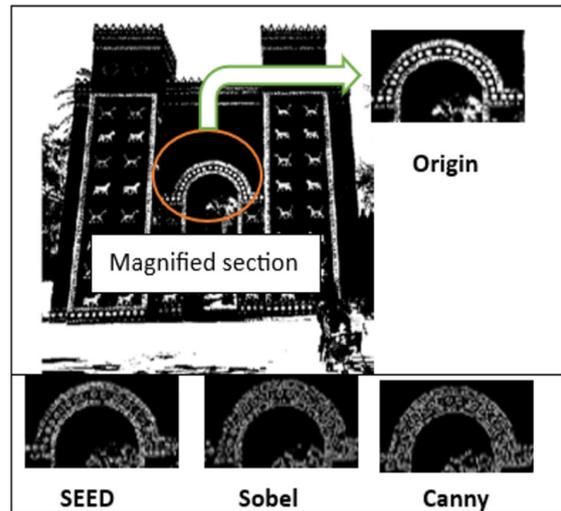


Fig.10: Searching for False Edges in 'Ishtar Gate'.

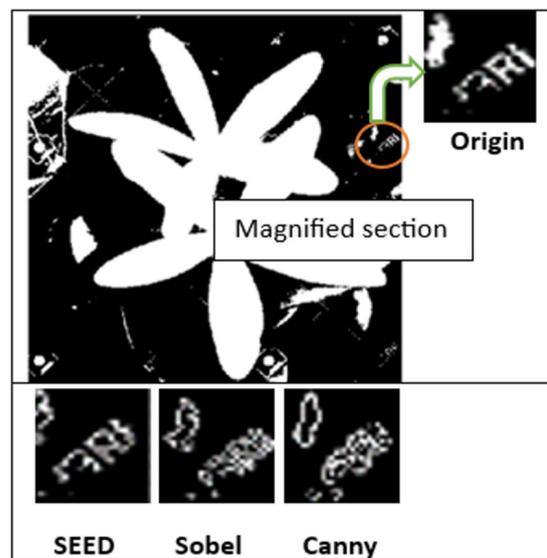


Fig.11: Searching for False Edges in 'Flowers'.

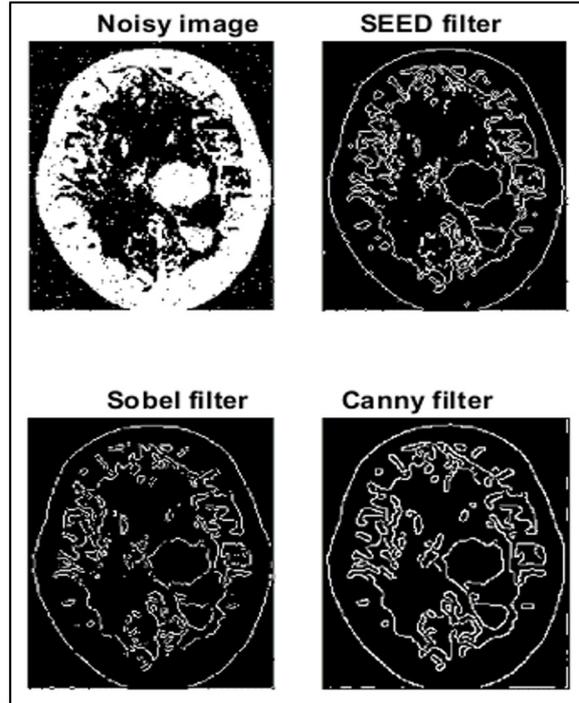
**Table 2:** Jaccard Index (IoU) for Noise-Free Images.

Image	SEED	Sobel	Canny
Camera Man	0.1698	0.0629	0.0771
CT-Scan	0.2358	0.0989	0.1005
Ishtar Gate	0.1869	0.0710	0.0653
Flowers	0.0776	0.0361	0.0376

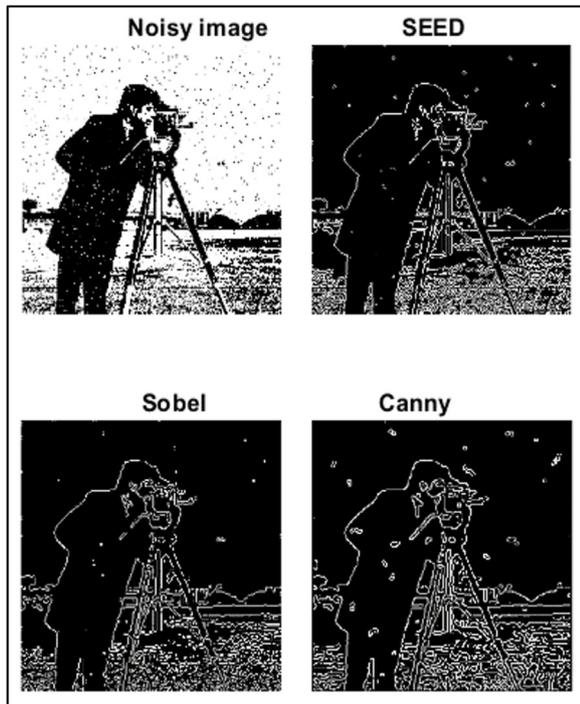
the Sobel and Canny algorithms.

The second set of images uses the images of the first set with the addition of 3% “salt and pepper” noise. In this test, the first phase of the SEED algorithm has been activated, followed by the edge detection phase. Fig. 12 to Fig. 15 presented the results.

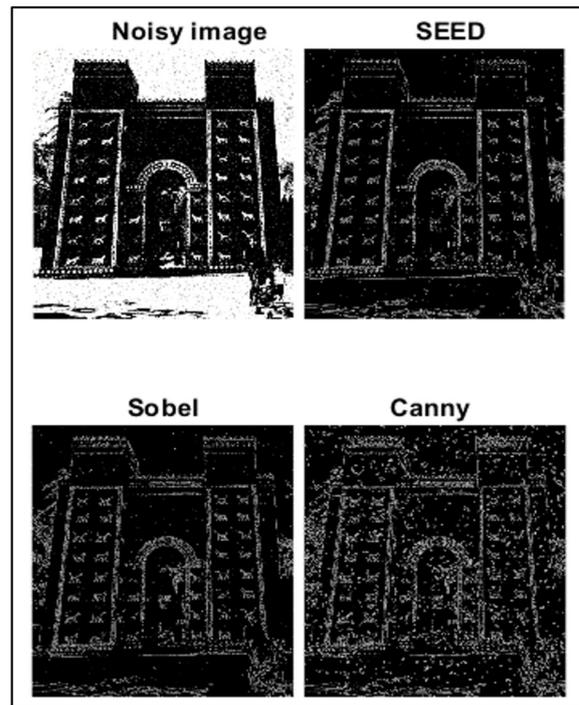
In a qualitative assessment, examining of the figures representing the edges of the noisy images shows that both the ‘SEED’ and ‘Sobel’ algorithms typically outperform the ‘Canny’ algorithm. Table 3 includes the calculated PSNR and MSE values from which a quantitative evaluation can be made. In this evaluation, the SEED algorithm once again registered the highest PSNR and the lowest MSE compared to the other two algorithms.



**Fig.13:** Edge Detection of Noisy ‘CT-Scan’.



**Fig.12:** Edge Detection of Noisy ‘Camera Man’.

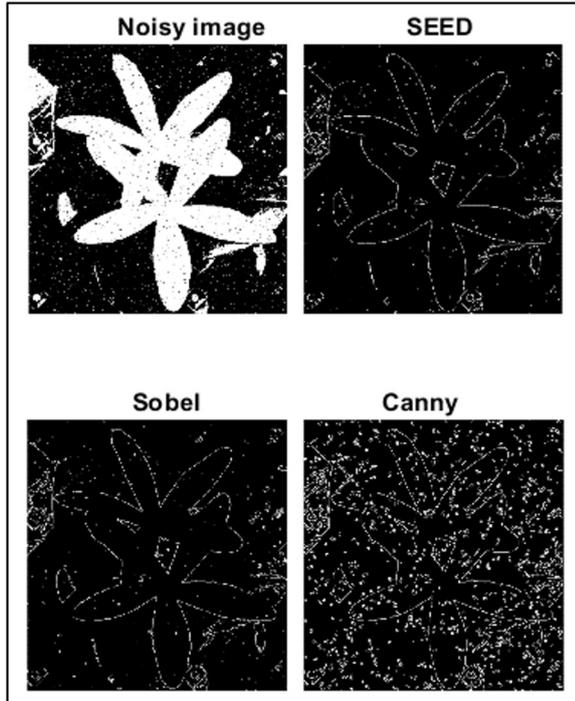


**Fig.14:** Edge Detection of Noisy ‘Ishtar Gate’.

The Jaccard index results for the noisy image set, as presented in Table 4, demonstrate that the SEED algorithm outperforms both the Sobel and Canny algorithms.

**Table 3:** PSNR and MSE for Noisy Images.

Image	PSNR			MSE		
	SEED	Sobel	Canny	SEED	Sobel	Canny
Camera Man	2.77	1.99	2.05	0.52	0.63	0.62
CT-Scan	4.75	3.63	3.51	0.33	0.43	0.44
Ishtar Gate	5.19	4.23	4.01	0.30	0.37	0.39
Flowers	4.94	4.52	4.30	0.32	0.35	0.37



**Fig.15:** Edge Detection of Noisy ‘Flowers’.

**Table 4:** Jaccard Index (IoU) for noisy Images.

Image	SEED	Sobel	Canny
Camera Man	0.1706	0.0653	0.0843
CT-Scan	0.2431	0.0953	0.1022
Ishtar Gate	0.1856	0.0732	0.0855
Flowers	0.0843	0.0364	0.0702

#### 4. CONCLUSION

In this era, the field of computer vision and image processing is gaining increasing importance. To understand images and recognize their details, components, and objects they contain, the need arises to use edge detection techniques, one of the most fundamental problems in the field of computer vision and image processing. This research introduced a Simple and Effective Edge Detection (SEED) algorithm, a novel approach leveraging Boolean logic operations to enhance edge detection in binary images. The experimental results highlighted the efficiency of SEED compared to commonly used methodologies, such as the Sobel and Canny methods. The main advantage of SEED is its computational efficiency based on Boolean operations, which are processed quickly by analyzing successive pairs of pixels. SEED was able to overcome and, in some cases, process noise, which resulted in a significant reduction in false edge detection.

Moreover, the proposed algorithm obtains thin edges, with one pixel wide, for the processed images. Slimming edges reduces the amount of data to process and puts the focus on the data that represents

the actual edges. An important metric in evaluating the effectiveness of edge detection algorithms, especially on binary images, is the intersection over union (IoU). Here, it is necessary to confirm that the thickness of the borders has an impact on the outputs of the IoU values. Even if edge locations are captured accurately, thicker edges may negatively affect IoU results. In conclusion, in addition to the improved accuracy and quality provides in detecting edges, the SEED algorithm can open the way to more efficient and powerful techniques in the field of dealing with computer vision and digital image processing.

#### ACKNOWLEDGEMENT

We acknowledge the Mustansiriyah University ([www.uomustansiriyah.edu.iq](http://www.uomustansiriyah.edu.iq)) for supporting this work.

#### AUTHOR CONTRIBUTIONS

Conceptualization, M.A. Tawfeeq and M.Z. Abdallah; methodology, M.A. Tawfeeq and M.Z. Abdallah; software, M.A. Tawfeeq; validation, M.A. Tawfeeq and M.Z. Abdallah; formal analysis, M.A. Tawfeeq; investigation, M.A. Tawfeeq and M.Z. Abdallah; data curation, M.A. Tawfeeq and M.Z. Abdallah; writing—original draft preparation, M.A. Tawfeeq and M.Z. Abdallah; writing—review and editing, M.A. Tawfeeq and M.Z. Abdallah; visualization, M.A. Tawfeeq and M.Z. Abdallah; supervision, M.A. Tawfeeq; funding acquisition, M.A. Tawfeeq and M.Z. Abdallah. All authors have read and agreed to the published version of the manuscript.

#### References

- [1] B. Tian and W. Wei, “Research Overview on Edge Detection Algorithms Based on Deep Learning and Image Fusion,” *Security and Communication Networks*, 2022(Article ID 1155814), 2022.
- [2] T. Lu, B. Zhang, Y. Hu, and J. Chen, “Computed Tomography Imaging Based on Edge Detection Algorithm in Diagnosis and Rehabilitation Nursing of Stroke Patients with Motor Dysfunction,” *Scientific Programming*, 2021(Article ID 5499351), ten pages, 2021.
- [3] M. H. Kolekar, *Intelligent Video Surveillance Systems: An Algorithmic Approach*, Chapman and Hall/CRC. 2018.
- [4] Dharampal and V. Mutneja, “Methods of Image Edge Detection: A Review,” *Journal of Electrical & Electron Systems*, vol. 4, no. 150, 2015.
- [5] S. Das S, “Comparison of various edge detection technique,” *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 9, no. 2, pp. 143-158, 2016.
- [6] M. A. Ansari, D. Kurchaniya and M. Dixit, “A Comprehensive Analysis of Image Edge Detec-

- tion Techniques,” *International Journal of Multimedia and Ubiquitous Engineering*, vol. 12, no. 11, pp.1-12, 2017.
- [7] A. K. Bharodiya and A. M. Gonsai, “An improved edge detection algorithm for X-Ray images based on the statistical range,” *Heliyon*, vol. 5, no. 10, 2019.
- [8] K. M. R. A. Utama, R. Umar, and A. Yudhana, “Edge detection comparative analysis using Roberts, Sobel, Prewitt, and Canny methods,” *Journal Teknologi dan Sistem Komputer*, vol. 10, no. 2, pp. 67-71, 2022.
- [9] Q. Yu, H. Tang, K. C. Tan, and H. Yu, “A brain-inspired spiking neural network model with temporal encoding and learning,” *Neurocomputing*, vol. 138, pp. 3–13, 2014.
- [10] H. Yedjour, B. Mefta, O. Ležoray, and A. Benyettou, “Edge detection based on Hodgkin–Huxley neuron model simulation,” *Cognitive Processing*, vol. 8, pp. 315–323, 2017.
- [11] V. Demin and D.Nekhaev, “Recurrent Spiking Neural Network Learning Based on a Competitive Maximization of Neuronal Activity,” *Frontiers in Neuroinformatics*, vol. 12, No. 79, 2018.
- [12] G. Bellec, D. Sala, A. Subramoney, R. A. Legenstein, and W. Maass, “Long short-term memory and learning-to-learn in networks of spiking neurons,” in *Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS 2018)*, Montreal, QC, Canada, 3–8 December 2018.
- [13] K.V. Vemuru, “Image Edge Detector with Gabor Type Filters Using a Spiking Neural Network of Biologically Inspired Neurons,” *Algorithms*, vol. 13, no. 165, 2020.
- [14] H. Wang, B. Yu, K. Xia, J. Li, and X. Zuo, “Skeleton edge motion networks for human action recognition,” *Neurocomputing*, vol. 423, pp. 1–12, 2021.
- [15] S. M. Avendano, D. M. Parra, and O. R. Salguero, “Triangle mesh skeletonization using non-deterministic voxel thinning and graph spectrum segmentation,” *MATEC Web of Conf.*, vol. 336, 2021.
- [16] L. Ren, X. Wang, N. Wang, G. Zhang, Y. Li, and Z. Yang, “An edge thinning algorithm based on newly defined single-pixel edge patterns,” *IET Image Processing*, vol. 17, no.4, pp. 1161-1169, 2023.
- [17] X. Li, H. Jiao H., and Y. Wang, “Edge detection algorithm of cancer image based on deep learning,” *Bioengineered*, vol.11, no. 1, pp. 693-707, 2020.
- [18] J. Chen, C. Du, Y. Zhang, P. Han, and W. Wei, “A clustering-based coverage path planning method for autonomous heterogeneous UAVs,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 25546-25556, 2022.
- [19] Y. Liu and J. Zhu, “Sub-pixel edge detection algorithm of spot image based on improved Zernike moment,” *Laser Journal*, vol. 42, no. 5, pp. 32–35, 2021.
- [20] H. Un, X. Hao, J. Wang, B. Pan, P. Pei, B. Tai, Y. Zhao, and S. Feng, “Flame Edge Detection Method Based on a Convolutional Neural Network,” *ACS Omega*, vol. 7, no. 30, pp. 26680-26686, 2022.
- [21] W. Guohua, S. Zhixiong, H. Min, and Y. Longfei, “Edge detection in single multimode fiber imaging based on deep learning,” *Optics Express*, vol. 30, no. 17, pp. 30718-30726, 2022.
- [22] Y. Jiahao, Z. Li, L. Xiaohui, P. Hong, and W. Jun, “A novel edge detection method based on dynamic threshold neural P systems with orientation,” *Digital Signal Processing*, vol. 127, Iss. 103526, 2022.
- [23] X. Ronghao, L. Rikong, P. Hong, Y. Qian, L. Xiaohui, and W. Jun, “Edge Detection Method Based on Nonlinear Spiking Neural Systems,” *International Journal of Neural Systems*, vol. 33, no. 1, 2023.
- [24] J. Al-Azzeh, B. Zahran, and Z. Alqadi, “Salt and Pepper Noise: Effects and Removal,” *International Journal on Informatics Visualization*, vol. 2, no. 4, 2018.



**Mohammed Ali Tawfeeq** is a Professor at Computer Engineering Department, College of Engineering, Mustansiriyah University. He received his B.Sc. in Electrical Engineering from University of Technology/Baghdad, Iraq in 1979 and his M.Sc. degree in Computer Engineering from University of Baghdad in 1989. He obtained his Ph.D. in Computer Engineering from University of Technology, Baghdad, Iraq in 2006.

His research interests include but limited to, Intelligent Systems, Optimization Techniques, Wireless Sensor Networks, Deep Learning, and Image processing.



**Mahmood Zaki Abdullah** was born in Baghdad - Iraq on 6th March 1969. He is currently an Assistant Professor in the Computer Engineering Department at the College of Engineering - Mustansiriyah University. He got a PhD in 2007 and MSc degrees in 2002 in Computer Engineering from the University of Technology, and a BSc degree from the University of Baghdad in 1991.

The research interests include information technology, software engineering, computer networks, Intelligent systems, and all specializations related to computer engineering.