IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 1 July 2024, accepted 24 July 2024, date of publication 1 August 2024, date of current version 13 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3436661

RESEARCH ARTICLE

An Integrated Method Using Linear Homotopy Parametric Value for Enhancing Image Edge Detection

S. SHIVAM KUMAR JHA AND N. MOHANA
Department of Mathematics, School of Advanced Sciences, Vellore Institute of Technology, Chennai 600127, India
Corresponding author: N. Mohana (mohana.n@vit.ac.in)

ABSTRACT This study introduces an integrated convolutional mask method for enhancing edge detection in digital images by combining the linear homotopy parametric value (LHP) with classical edge detection operators. Our results, obtained from extensive experimentation and comparison with conventional MATLAB-based methods, demonstrate that the LHP-based method outperforms previous techniques in terms of reliability and structural accuracy. LHP is integrated to address issues of noise sensitivity and diverse image structures, highlighting the algorithm's improved performance across a range of images.

INDEX TERMS Convolution, edge detection, enhancement, gradient based method, LHP.

I. INTRODUCTION

A fundamental technique in image processing, edge detection is essential for computer vision and image analysis. It is designed to identify boundaries within an image by emphasizing regions where significant variations in color or intensity occur. This procedure is crucial for tasks such as feature extraction, image segmentation, and object recognition [4]. While edge enhancement involves increasing the contrast between the edge and background to make the edge more noticeable, edge detection focuses on locating edge pixels. Numerous edge detection methods and algorithms exist, each with unique advantages and disadvantages. Common techniques include the Canny edge detector, the Laplacian of Gaussian (LoG) method [7], and gradient-based techniques such as Sobel, Prewitt, and Roberts operators, among others.

Distinguishing edges can be challenging in the presence of noise, poor contrast, or complex textures. Our MATLAB-based approach focuses on enhancing the precision and quality of the identified edges, which can be influenced by parameters and mask selection. To achieve the best comparison results with the classical Canny edge detection method available in MATLAB, we have also applied this method to various other operators, such as LoG, Sobel, Prewitt, and Roberts, with Canny outperforming the rest.

In grayscale images, pixel values range from 0 to 255, with each pixel's brightness directly indicated by its value. Grayscale images have only one channel for intensity, whereas RGB images are separated into different channels for color. Specifically, we focus on enhancing image quality by optimizing parameters to balance blur, brightness, and contrast. Finally, we include the metric values of our analysis and examples of output images from the original, Canny, and our proposed method, demonstrating the advantages of using our approach.

A. QUALITY FACTORS AND TECHNIQUES IN EDGE DETECTION

A comprehensive edge detection algorithm is designed by integrating contrast [8], correlation, energy, entropy, and homogeneity, with similar work found in [9]. The difference in intensity between neighboring pixels is explored as a quality factor influencing edge detection. We explore contrast-based edge enhancement techniques to improve the visibility of edges against varying backgrounds and leverage correlation-based approaches indicating pixel pattern

The associate editor coordinating the review of this manuscript and approving it for publication was Yongjie Li.

similarity to refine edge detection. We compare the efficacy of these methods with traditional techniques for capturing patterns and edges. Additionally, measures of intensity distribution are integrated into edge detection algorithms. By considering energy, we aim to enhance the robustness of edge detection, particularly in images with varying levels of intensity. Using entropy [11], a measure of uncertainty or disorder, we improve edge identification in noise or texture-filled images. We discuss entropy-based methods and their effectiveness in challenging visual settings for locating edges. We also examine homogeneity as a quality factor for edge detection, which denotes pixel uniformity. A survey on image detection effectively outlines the approach and earlier techniques [13]. In this paper, we present an enhanced version of the Canny edge detector, as well as gradient-based techniques such as the Roberts operator, Prewitt operator, and Sobel operator.

B. CANNY EDGE DETECTOR

The Canny edge detector is one of the conventional methods for edge detection. It remains the most efficient operator with the greatest number of variations, having been first proposed by J. Canny for his M.Sc. thesis at MIT in 1983 [10]. When searching for image edges using the technique of isolating noise from the image, the Canny approach becomes even more important. Without changing the image's edge properties, this method is more effective after applying the appropriate threshold value, showing a strong tendency to locate edges. Our method integrates with the classical Canny edge operator method to provide an enhanced version of Canny. The method consists of several steps, each intended to improve and enhance the edge detection technique. Related work on the upgrade of the Canny operator can be found in [12]. In our method, we apply Gaussian blur, and then the gradient computed using convolution indicates potential edges. Non-maximum suppression results in a thinner and more accurate representation of edges. Two thresholds are applied to categorize edges, enabling the incorporation of hysteresis for final Canny edge detection. Finally, we apply the LHP mask to enhance the outcome of the Canny edge operator.

C. GRADIENT BASED METHODS

For the three most common operators, Marr and Hildreth [5], Prewitt [6], and Sobel and Feldman [1], [2], we will apply our LHP method to achieve improved edge detection images. These improved images are significantly better than those produced by the conventional methods, showcasing how our proposed approach surpasses them with more vivid and clear outcomes. All these operators employ small 2 × 2 or pairs of 3 × 3 convolution kernels for edge detection, with one kernel for detecting vertical edges (P_x) and the other for horizontal edges (P_y). Traditional Robert masks use a 2 × 2 matrix

$$R_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \; R_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

where as the Prewitt and sobel uses 3 × 3 matrix with

$$P_x = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}, \; P_y = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

and $S_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$, S_y as the transpose of S_x.

Here,

$$\mu_R = ({R_x}^2 + {R_y}^2)^{1/2},$$
$$\mu_P = ({P_x}^2 + {P_y}^2)^{1/2},$$
$$\mu_S = ({S_x}^2 + {S_y}^2)^{1/2},$$

where μ_R is the gradient magnitude of the Robert operator, μ_P is the gradient magnitude of the Prewitt operator, and μ_S is the gradient magnitude of the Sobel operator. Here, our dedicated mask with the average value integrates with these older masks, and by using the parameters, we achieve a better-enhanced image of the detected edges. The convolution of an image with a Gaussian smoothing filter followed by the Laplacian filter constitutes the Laplacian of Gaussian (LoG) image processing operation. This operation aims to enhance an image's edges and reduce noise. The Laplacian operator highlights regions of abrupt intensity change, while Gaussian smoothing helps to reduce noise sensitivity.

II. METHODOLOGY

Many methods for detecting edges were explored, and while the discussed methods are diverse, some are effective in detecting edges of the overall picture but not in unimproved subregions. To address this, the enhanced edge detection model suggested in this study is based on a new class of homotopy from emerging algebraic topology. Compared to traditional image edge detection models, this approach offers an improved version of the image and provides a better display of image information. Homotopy is one of the most powerful invariants of algebraic topology, and when linearity is imposed on these classical invariants through the concatenation of paths, it enables many modern applications. Here, we use one of the factors to compute the LHP values, which include α^*, β^* and γ^*.

A path in space X that we are relating with loops here is nothing but a continuous mapping over a unit period of time such that $f : I \to X$ where $I = [0, 1]$. A homotopy of paths in X is a family $f_t : I \to X, \; 0 \le t \le 1$, such that the endpoints $f_t(0) = x_0$ and $f_t(1) = x_1$ are independent of t and the associated map $F : I \times I \to X$ defined by $F(x, t) = f_t(x)$ is continuous [3].

If two paths f_0 and f_1 are connected in the aforementioned way of homotopy f_t, then they are said to be homotopic paths, i.e., $f_0 \simeq f_1$. Essentially, any two paths in $\mathbb{R}^n$ with the same initial or endpoints x_0 and x_1 are homotopic via the homotopy:

$$f_t(x) = (1 - t)f_0(x) + tf_1(x), \tag{1}$$

where each point $f_0(x)$ travels at a constant speed along the line segment to $f_1(x)$. One can think of an image's edges as continuous paths. The homotopy concept ensures that the detected edges create continuous, smooth transitions, maintaining the integrity of the shapes and borders in the image.

If $\alpha, \beta, \gamma : I \to X$ are paths such that $\alpha(1) = \beta(0)$ and $\beta(1) = \gamma(0)$, then $\alpha \circ \beta \circ \gamma : I \to X$ is a composition or product path which traverses from α to γ as

$$\alpha \circ \beta \circ \gamma(x) = \begin{cases} \alpha(3x), & \text{whenever } 0 \le x \le 1/3 \\ \beta(3x-1), & \text{whenever } 1/3 \le x \le 2/3 \\ \gamma(3x-2), & \text{whenever } 2/3 \le x \le 1 \end{cases} \tag{2}$$

Paths α, β, and γ represent segments of a continuous path in a topological space. These paths are concatenated to form a single path from one point to another, capturing transitions and changes in direction, as shown in Figure 1. The function $\alpha \circ \beta \circ \gamma(x)$ is defined piecewise over the interval $[0, 1]$, allowing the path to change its rule at specific points. This models edge detection in images, where different segments might exhibit different edge characteristics. The piecewise function uses different rules (paths) for different segments to model these variations. From the above two equations (1) and (2), we have

$$\begin{aligned} f_t(x) &= (1-t)\alpha(3x) + t\beta(3x-1) + (1-t)\beta(3x-1) \\ &\quad + t\gamma(3x-2) + (1-t)\gamma(3x-2) + t\alpha(3x) \\ &= \alpha(3x) + \beta(3x) + t\alpha(3x) - \beta(1) + t\gamma(3x) - t\gamma(2)\,. \end{aligned}$$

When the path is closed, we add $(1-t)\gamma(3x-2) + t\alpha(3x)$ that is, $\alpha(0) = \gamma(1)$ gives

$$f_t(x) = \alpha(3x) + \beta(3x) + \gamma(3x) - \beta(1) - \gamma(2)\,,$$

our equation becomes independent of t. Therefore, there won't be any changes as a result of t; hence, in our case, $f_t(x) = f(x)$.

$$f(x) = \alpha(3x) + \beta(3x) + \gamma(3x) - \beta(1) - \gamma(2)\,. \tag{3}$$

Constants like α^*, β^*, and γ^* are introduced to scale the paths correctly. These parameters adjust the sensitivity and accuracy of edge detection algorithms. Simplifying the combined effect of these paths into a single function allows for efficient computation of edges, effectively highlighting areas of high gradient (edges). In addition, we have our parametric values in a constant function as

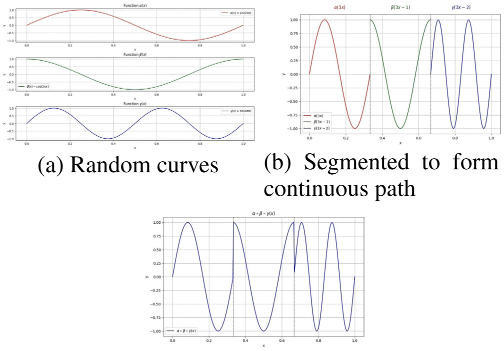
$$\alpha^* + \beta^* + \gamma^* = f(5) \tag{4}$$

which statisfies our model, additionally setting

$$\alpha(x) = x/3, \quad \beta(x) = (1-x)/3, \quad \gamma(x) = x/5\,.$$

Substituting the above equation in equation(3), we have

$$\begin{aligned} f(x) = &\ 3\alpha(x) + 3\beta(x) + 3\gamma(x) - \beta(1) - \gamma(2) \\ = &\ 1 + (3x)/5 - 2/5 \\ = &\ 3\,((1+x)/5) \end{aligned}$$



(a) Random curves

(b) Segmented to form a continuous path

(c) Paths are concatenated to form a single path

FIGURE 1. Concatenation of paths.

which becomes $f(5) = 36/10$, for $x = 5$, from equation (4) we can conclude that

$$\alpha^* + \beta^* + \gamma^* = 36/10\,.$$

Parametric values and piecewise functions help in fine-tuning the edge detection process, ensuring that different parts of the image are analyzed appropriately. The range of α^*, β^*, and γ^* lies between $[1, 2]$ to achieve better enhancement output for this dedicated mask. One of the hardest challenges in the edge detection process is quantitative evaluation. A single, universal metric can't be used to evaluate an enhanced algorithm's efficiency. Given the variations in individual visual perception, a quantitative analysis is required to verify the subjective evaluation of the enhanced pictures. The five different quantitative indicators used to evaluate the enhancement quality throughout this paper are contrast, correlation, energy, homogeneity, and entropy. Additionally, we will determine the Optimal Dataset Scale (ODS), Optimal Image Scale (OIS), and Average Precision (AP). In our case, the binary image will be from one of the five operators, which will be taken as the ground truth image to compute our ODS, OIS, and AP. The parametric values α^*, β^*, and γ^* are used to improve the image. We have denoted I_E and M as the enhanced edge detected image and mask for LHP respectively. Let I_p denote the input image with

$$I_E(x, y) = M * I_p(x, y)\,,$$

where $*$ is the convolution product. The parametric values α^*, β^*, and γ^* represent the mask of the LHP window. Based on our parametric values, we have diligently utilized a specific mask through extensive trial and error, which has significantly enhanced our method, table 1 shows how eight masks are made using distinct directions from this investigation. For this method, when we try to do the same for other directional values, we end up with sometimes over-exposed or under-exposed images, which makes them look unnatural, along with less coverage in the image boundaries of moderate quality.

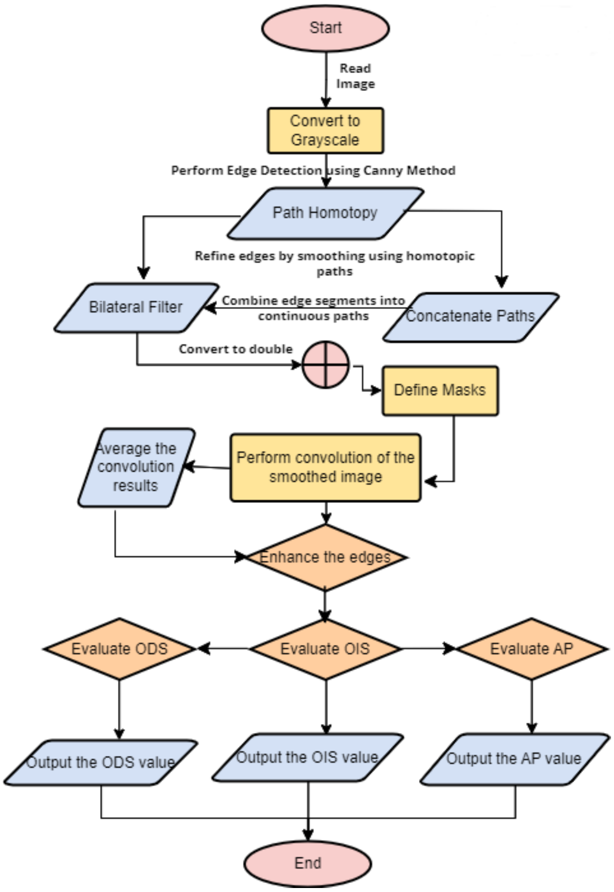


FIGURE 2. Flowchart of LHP.

Our working model is illustrated in the flowchart in Figure 2. In our method, the original RGB image is first converted to grayscale because edge detection algorithms typically operate on single-channel images. The conversion is done using a weighted sum of the RGB components, typically using the formula:

$$\text{Grayscale} = 0.299\text{R} + 0.587\text{G} + 0.114\text{B}$$

Then edge detection is performed using traditional operators like Canny, Sobel, or Prewitt. These algorithms calculate the gradient of the image intensity at each pixel, highlighting regions with high spatial derivatives. The bilateral filter is an edge-preserving smoothing filter that combines domain and range filtering to preserve edges while reducing noise. it's represented as:

$$I'(x) = \frac{1}{W_p} \sum_{x_i \in S} I(x_i) f_r(||I(x_i) - I(x)||) g_s(||x_i - x||)$$

where I is the input image, I' is the output image, f_r is the range kernel for intensity differences, g_s is the spatial kernel for coordinate differences, and W_p is a normalization factor. The bilateral filter balances both spatial proximity (controlled by g_s) and intensity similarity (controlled by f_r) to achieve edge-preserving smoothing while reducing noise in the output image.

TABLE 1. Dedicated directional convolutional mask suitable for our parametric values.

$\begin{bmatrix} 1 & 0 & 0 \\ \gamma^* & \beta^* & \alpha^* \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & \gamma^* & 0 \\ 0 & \beta^* & 0 \\ 0 & \alpha^* & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ -2 & \beta^* & \gamma^* \\ 2 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & \alpha^* & 0 \\ 0 & \beta^* & 0 \\ 2 & \gamma^* & 1 \end{bmatrix}$
$\begin{bmatrix} 0 & 0 & \gamma^* \\ 0 & \beta^* & 0 \\ \alpha^* & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} \gamma^* & 0 & 0 \\ 0 & \beta^* & 0 \\ 0 & 0 & \alpha^* \end{bmatrix}$	$\begin{bmatrix} \alpha^* & 0 & 0 \\ 0 & \beta^* & 0 \\ -2 & 0 & \gamma^* \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & \alpha^* \\ 0 & \beta^* & 0 \\ \gamma^* & 0 & 0 \end{bmatrix}$

A. DEDICATED MASKS

Two functions (or images) are combined by the convolution technique to create a third function (or image). In order to calculate the weighted total of the overlapping values of two functions, one (the filter or mask) must be slid over the other (the input image). Usually, the output image, or new function created by the convolution, retains characteristics like edges, textures, and patterns. With regard to identifying edges or other features, every mask corresponds to a certain orientation. Upon convolving the input image with these masks, the weighted total of the pixel values in the overlapping regions is calculated, leading to the output image highlighting distinct features. Here, the image is subjected to a variety of filters (masks) via convolution. For edge detection, every mask corresponds to a distinct orientation. The general convolution operation is defined as:

$$C(x, y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} H(i, j) \cdot I(x - i, y - j)$$

where C is the output image, H is the filter mask, and I is the input image. The mask in the table 1 shows how we carefully adjust the settings to obtain several orientations that are well suited to our parameters and methodology. Out of the eight conventional masks, we have taken the average mask of all eight directional masks that are appropriate and provide specific results.

B. PERFOMANCE METRICS

Optimal Dataset Scale (ODS) [14]: This metric evaluates the overall performance across all images in a dataset at a fixed scale (threshold = 1). It's calculated using precision and recall, which are derived from true positives (TP), false positives (FP), and false negatives (FN)

$$\text{precision} = (\text{TP})/(\text{TP} + \text{FP})$$

$$\text{recall} = (\text{TP})/(\text{TP} + \text{FN})$$

$$\text{ODS} = (2 \cdot \text{precision} \cdot \text{recall})/(\text{precision} + \text{recall})$$

Optimal Image Scale (OIS) [14]: This metric evaluates the best possible performance for each image individually, then averages the results. It's also based on the F1 score, which is the harmonic mean of precision and recall.

Average Precision (AP): This metric summarizes the precision-recall curve as the weighted mean of precisions achieved at each threshold, with the increase in recall from the previous threshold used as the weight

$$\text{AP} = \sum_{n} (R_n - R_{n-1}) P_n$$

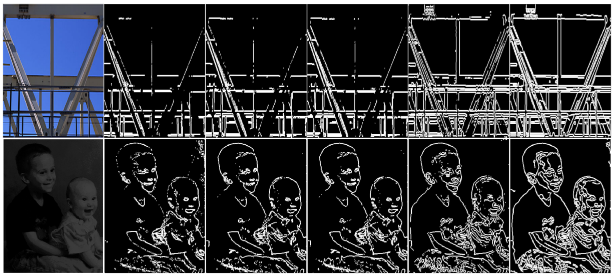


FIGURE 3. LHP applied on Robert's, Prewitt, Sobel, Log and Canny operator.

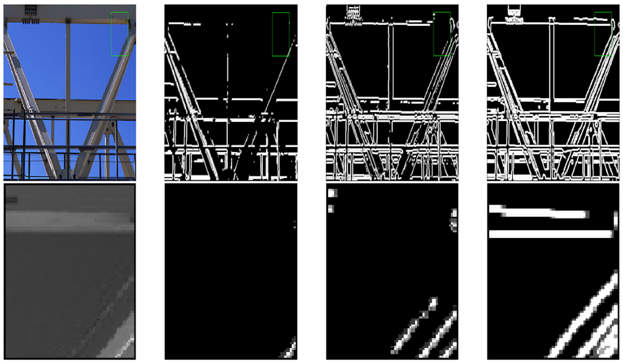


FIGURE 4. Scaled section of the gantrycrane image displaying the Sobel, LoG, and Canny regions that were found.

where P_n and R_n are the precision and recall at the nth threshold.

III. EXPERIMENTAL RESULTS

We have selected a 3 × 3-pixel window, which is theoretically better than a 5 × 5-pixel window or larger, as the latter may require more metric parametric values to achieve nearly the same result or may cause an image to appear excessively exposed or blurry. The built-in images in MATLAB, resized to 300 × 400, are utilized in this article. The edge detection measurements of the proposed algorithm aim to improve the edge-detected regions by displaying high-quality metric values such as ODS, OIS, and AP, including contrast, correlation, energy, entropy, and homogeneity. These metrics assess the effectiveness of the edge detection technique and compare the enhanced image quality to the original. Once the setup is done, our suggested method takes only a few seconds per image test. The performance test was implemented using MATLAB(Mathworks) on Windows 11. Processor: 11th Gen Intel(R) Core(TM) i3-1115G4 @ 3.00GHz. This method works very well and surpasses the traditional methods available in MATLAB. Additionally, this method's primary benefit is its ability to capture low-light image edges in a way that other methods cannot. This method brightens and broadens edges, making the edge-detected image appear perfect to the human eye. However, it also blurs edges to connect broken lines by filling in the spaces.

The method we propose improves the total number of edges detected by applying all five operators. Among them, LHP applied to Canny, denoted as LHP_C, provides superior results, while LHP applied to Log yields better results for images with less noise or detail. Figure 3 illustrates the application of several operators using our method by highlighting the edge area of a color image, including Robert, Prewitt, Sobel, Log, and Canny. In this case, Canny and Log perform better than the other operators, which only capture the outer pixel difference and miss inner boundaries. The figure5 and figure6 demonstrate the LHP applied to the Canny operator on various grayscale and color images available on MATLAB; the first row displays the original image, while the second row displays the images produced by the Canny operator, and the third row displays the enhanced edge detected images using LHP_C. The pixel difference nearer the inner boundary is captured by our suggested method, as seen in figures 3 and 4, while the other operators were unable to do so. To introduce readers to the values,

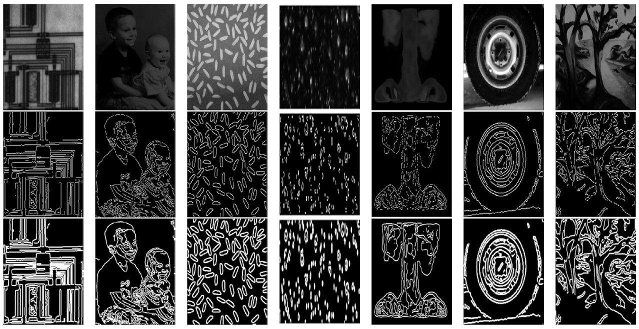


FIGURE 5. LHP_C applied on grayscale images; first row: original images, second row: canny algorithm based images, and third row: LHP based images.

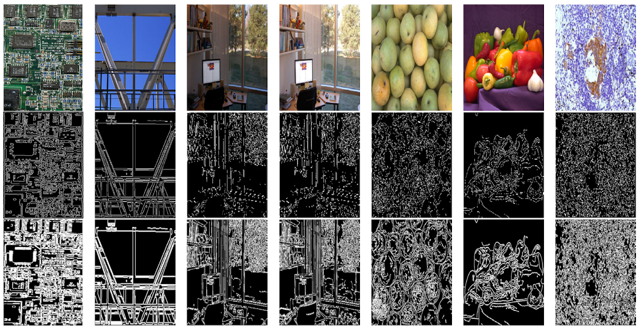


FIGURE 6. LHP_C applied on color images; first row: original images, second row: canny algorithm based images, and third row: LHP based images.

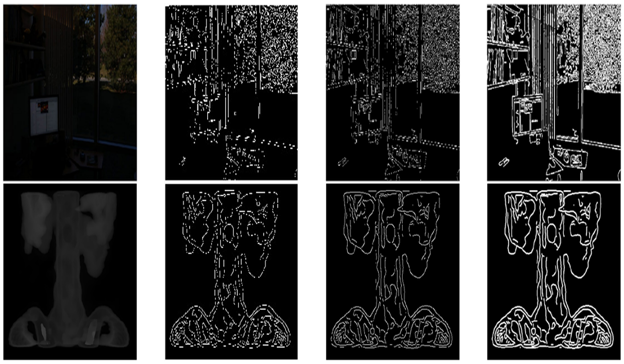


FIGURE 7. A random comparison of both color and grayscale images of original, Canny, bilateral Canny using LHP and LHP_C.

TABLE 2. Computational values of various quality factors.

	Contrast		Correlation		Energy	
Images	**Canny**	**LHP_C**	**Canny**	**LHP_C**	**Canny**	**LHP_C**
Gantrycrane	0.0880	2.3295	0.1821	0.8632	0.7358	0.4517
Peppers	0.0659	1.8680	0.4811	0.8467	0.8115	0.5672
Board	0.2083	4.5649	0.3537	0.7778	0.5128	0.2036
Trees	0.0884	2.4413	0.4605	0.8366	0.7554	0.4648
Rice	0.1120	3.3238	0.3512	0.7798	0.7279	0.4287
Spine	0.0522	1.6512	0.4273	0.8226	0.8593	0.6747

TABLE 3. Homogeneity and entropy values obtained using LHP_C.

	Homogeneity		Entropy	
Images	**Canny**	**LHP_C**	**Canny**	**LHP_C**
Gantrycrane	0.9560	0.8822	0.47598	1.7294
Peppers	0.9671	0.8889	0.35831	1.4612
Board	0.895	0.7640	0.72386	2.5213
Trees	0.9558	0.8559	0.43592	1.7583
Rice	0.9440	0.8246	0.4531	1.8168
Spine	0.9739	0.9162	0.27693	1.0911

TABLE 4. Perfomance metrics.

	Proposed method LHP_C		
Images	**ODS**	**OIS**	**AP**
Gantrycrane	0.9445	0.9175	0.9982
Peppers	0.9364	0.8915	0.9983
Board	0.9316	0.9441	0.9893
Trees	0.9383	0.8860	0.9987
Rice	0.9218	0.8681	0.9993
Spine	0.9409	0.8594	0.9990

we have included the experimental computational values of the different quality factors in tables 2, 3, and 4. In addition, figure 7 illustrates the comparison between LHP_C and the original Canny algorithm.

IV. CONCLUSION

In conclusion, our proposed method demonstrates superior performance in edge detection when applied to the Canny operator, surpassing previous approaches and other MATLAB operators. By integrating the LHP method, we achieve advanced edge detection and image processing results. Our LHP-based method enhances crucial edges and outperforms traditional approaches, utilizing innovative algorithms to create more effective tools. We are committed to refining our approach to remain competitive with the latest methods, minimizing code and execution time while preserving image integrity. Future research will focus on persistent homology to determine the genus and improve edge detection, contributing significantly to tumor and cancer cell detection.

V. ACKNOWLEDGMENT

The authors would like to extend my heartfelt gratitude to Prof. Kalyani Desikan, Department of Mathematics, VIT Chennai, India, for her invaluable assistance in shaping the main draft of this article.

REFERENCES

[1] I. Sobel and G. Feldman, “A 3 × 3 isotropic gradient operator for image processing,” presented at the Stanford Artificial Project, 1968.

[2] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. Hoboken, NJ, USA: Wiley, 1973, pp. 271–272.

[3] A. Hatcher, *Algebraic Topology*. Cambridge, U.K.: Cambridge Univ. Press, 2002.

[4] P. Ganesan, V. Rajini, and R. I. Rajkumar, “Segmentation and edge detection of color images using CIELAB color space and edge detectors,” in *Proc. INTERACT*, Chennai, India, 2010, pp. 393–397, doi: 10.1109/INTERACT.2010.5706186.

[5] D. Marr and E. Hildreth, “Theory of edge detection,” *Proc. Roy. Soc. London B, Biol. Sci.*, vol. 207, no. 1167, pp. 187–217, 1980, doi: 10.1098/rspb.1980.0020.

[6] J. M. S. Prewitt, “Object enhancement and extraction,” in *Picture Processing and Psychopictorics*, B. Lipkin and A. Rosenfeld, Eds., New York, NY, USA: Academic, 1970, pp. 75–149.

[7] S. G. Mallat, “A theory for multiresolution signal decomposition: The wavelet representation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989, doi: 10.1109/34.192463.

[8] R. P. Johnson, “Contrast based edge detection,” *Pattern Recognit.*, vol. 23, nos. 3–4, pp. 311–318, Jan. 1990, doi: 10.1016/0031-3203(90)90018-g.

[9] C. Li, J. Zhou, and D. Dias, “Utilizing semantic-level computer vision for fracture trace characterization of hard rock pillars in underground space,” *Geosci. Frontiers*, vol. 15, no. 2, Mar. 2024, Art. no. 101769, doi: 10.1016/j.gsf.2023.101769.

[10] J. Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986, doi: 10.1109/tpami.1986.4767851.

[11] D. Sušanj, V. Tuhtan, L. Lenac, G. Gulan, I. Kožar, and Ž. Jericevic, “Using entropy information measures for edge detection in digital images,” in *Proc. 38th Int. Conv. Inf. Commun. Technol., Electron. Microelectron. (MIPRO)*, Opatija, Croatia, May 2015, pp. 352–355, doi: 10.1109/MIPRO.2015.7160293.

[12] C. Zhang, N. Zhang, W. Yu, S. Hu, X. Wang, and H. Liang, “Improved canny-based algorithm for image edge detection,” in *Proc. 36th Youth Academic Annu. Conf. Chin. Assoc. Autom. (YAC)*, Nanchang, China, May 2021, pp. 678–683, doi: 10.1109/YAC53711.2021.9486671.

[13] R. Sun, T. Lei, Q. Chen, Z. Wang, X. Du, W. Zhao, and A. K. Nandi, “Survey of image edge detection,” *Frontiers Signal Process.*, vol. 2, Mar. 2022, Art. no. 826967, doi: 10.3389/frsip.2022.826967.

[14] D. R. Martin, C. C. Fowlkes, and J. Malik, “Learning to detect natural image boundaries using local brightness, color, and texture cues,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530–549, May 2004, doi: 10.1109/TPAMI.2004.1273918.

S. SHIVAM KUMAR JHA received the B.Sc. and M.Sc. degrees in mathematics from Dwaraka Doss Goverdhan Doss Vaishnav College, University of Madras, in 2015 and 2018, respectively. He is currently pursuing the Ph.D. degree in algebraic geometry with Vellore Institute of Technology, Chennai, India. He presented a paper in the book of the proceedings of 2nd international conference on research trends in mathematics, in 2022.

N. MOHANA received the M.Sc. and M.Phil. degrees from the University of Madras, in 2010, and the Ph.D. degree from Vellore Institute of Technology (VIT), Chennai, India, in 2019. She has been an Assistant Professor with the Department of Mathematics, School of Advanced Sciences, VIT Chennai, since 2011. She has published more than 20 papers in reputed journals and conference proceedings. She has authored a book and also contributed to several book chapters. Her research interests include formal languages, automata theory, graph theory, and statistical analysis. She is a Lifelong Member of the Ramanujan Mathematical Society and the IAENG International Association of Engineers.