ASIAN JOURNAL OF INFORMATION TECHNOLOGY

MEDWELL PUBLICATIONS SINCE 2000

Edge Detection Methods in Scientific Cellular Photographs by Means of Analytical Assessment

Haeder Talib Mahde Alahmar
College of Engineering, Iraq University, Baghdad, Iraq

Key words: Night, edge detection, noise, Gauss, laplace, extract lasers, filter, coefficient, smoothing

Corresponding Author:
Haeder Talib Mahde Alahmar
College of Engineering, Iraq University, Baghdad, Iraq

Page No.: 95-101
Volume: 19, Issue 5, 2020
ISSN: 1682-3915
Asian Journal of Information Technology

Abstract: The normal scientific analysis of laboratory samples suffers from a lack of velocity and slowness in addition to the great effort exerted by way of the specialists which outcomes in different consequences for one sample in accordance to each specialist and his condition. In order to keep away from this on this approach is extraordinarily important in the cure of cellular scientific images. Edge detection minimizes the amount of information used. It ignores facts that does no longer work whilst retaining and manipulating the fundamental structural characteristics of the image. There is additionally a want to have an appropriate area detection concept. In this research, a range of strategies were used in the detection of the edges and their evaluation with extraordinary and utilized to a variety of samples of cell scientific images. This research was performed the usage of the MATLAB program. The end result used to be that the Laplacian of Gaussian coefficient had better consequences than almost all other algorithms in phrases of accuracy of the image.

INTRODUCTION

The classification of the cells in the blood images allows the evaluation and diagnosis of many diseases, such as cancer, malaria, leishmaniosis, anemia, etc., where the Samples are prepared and sent to the laboratory. If the specialist notices abnormal blood cells, something in the sample may indicate a disease or a condition. Visual examination by a blood specialist is tedious, stressful and time-consuming, so it is necessary to have a methodology and a system.

For automated testing of these blood samples, image processing techniques have been used to process medical images including techniques for detecting edges[1-4]. Edge detection is defined as the process of identifying and identifying sharp interruptions in images. Interference is a sudden change in pixel density that distinguishes borders for image purposes[5]. The traditional way of identifying edges involves manipulating and transforming the image using 2D filters that are built to be sensitive to large image changes and to return zero values in regular areas. There are many transactions available for edge detection, each designed to be sensitive to a specific type of edge. These parameters are selected depending on the edge orientation, noise environment and edge shape. The shape of the plants is a major aspect of the edges. So, that transactions can be improved in order to search for horizontal, vertical and sloping edges[6].

The edge detection process has difficulty in images with noise[7, 8], since, noise and edges have high frequency. The coefficients are large in images that are sufficiently noisy to do equality of average to reduce noise pixels but reduce the accuracy of the detection of the edges. All edges do not produce actual or direct intensity changes. Effects such as low light or focus for the same purposes, they are produced as incremental changes in density and therefore produce a threshold[5, 9]. Therefore, the coefficients must be well prepared to deal with such changes. What we have already concluded is that there are problems of misdiagnosis of the edges, finding false edges or even large computing time and other problems caused by noise and others. Therefore, there was a tendency to compare different techniques to discover the edges and load the performance of these techniques in different situation and conditions.

MATERIALS AND METHODS

Previous studies: The detection of the edges of many researchers is an important issue in the treatment of images in various areas as it is very important in the process of cutting and classification of images.

In the study[4], the researchers compared the detection techniques of the iris and the candidates for an algorithm to be adopted. In the study[3], researchers presented a variety of edge detection techniques with an assessment of their performance based on some known measures.

The study[8] proposes a two-phase noise-reduction scheme. Phase 1 uses an adaptive medium filter to determine the type of salt and pepper. The second phase is to be restored in a custom tuning mode applied only to pixels defined as noise and Keep edge, the result was 91% noise removal. The study[2] provides a discussion of the basic concepts of the various filters and applies these filters to the classification of sharks, depending on their species, so that, sharks are considered as a case study. In the study[10], the researchers present a new visual model of visual output in images derived from the principle of maximizing information and propose a new measure of visual output in pictures called the ambiguity site[11]. The researchers propose a new algorithm for cutting medical images based on noise removal and on the detection of edges for images.

The research objective: The detection of cellular medical images sometimes face problems such as detecting a false edge, missing the right edge, producing thick lines or low supply and problems caused by noise. Therefore, there was an analytical comparison between the techniques used in the detection of the edges after applied to samples of cellular medical images which were in our study are images of red blood cells infected with malaria parasites in order to access the best technique or algorithm in the detection of the edges of these images which gives the best results. In addition, to show the time of implementation of each algorithm and show the mystery of the resulting images apply the implementation of each algorithm[10].

Edge detection techniques and algorithms: There are many ways to implement the edge detection process but these methods can be grouped into two groups:

Slope-based edge detection techniques: The edges are exposed by looking for the largest and smallest value of the first derivative of the image.

Sobel coefficient: It is s a 3×3 filter composed of two nuclei, producing one nucleus from the other rotated at a 90° angle above equation sobel filter in x and y vector:

$$\begin{array}{ccc} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & \text{Gy} & -1 \end{array} \qquad \begin{array}{ccc} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & \text{Gx} \end{array}$$

Gx, Gy, shown in equation are the common masks used in the Sobel coefficient. These cores are designed to respond as far as possible to the vertically and horizontally spread edges of the pixel matrix. According to the direction of the nucleus, the nuclei can be applied separately to the input image in order to obtain the result in each direction (i.e., Gx, Gy). The separate values can then be combined to obtain an absolute value of the slope at each point and on the orientation of this inclination, 2 the slope value is given the following relationship:

$$|G| = \sqrt{Gx^2 + Gy^2} \tag{1}$$

The approximate value of the slope is calculated by using the following relationship:

$$|G| = \sqrt{Gx^2 + Gy^2} \tag{2}$$

Which is faster arithmetically and gives the angle of the edge direction (for the pixel grid) which causes the inclination to form:

$$\phi = \arctan(Gx/Gy) \tag{3}$$

Robert coefficient: This parameter is somewhat similar to the Sobel coefficient and calculatedthe slope in an image. The pixel value represents the absolute value of the input path at that point. The plants consist of two nuclei transformer each of them *22 as shown in equation. Each nucleus is produced by the other rotated 90° (Eq. 3).

The two nuclei/are designed to respond to the maximum potential of the 45° wide edges in the pixel matrix. So, that each nucleus corresponds to one of the orthogonal oblique trends in the pixel matrix. Any nucleus can be applied separately to the input image to find the slope values in each direction (Gy or Gx). Separate values can then be combined with discrete values to obtain absolute slope value at each point and slope direction. The slope value is given in the following relationship:

$$|G| = \sqrt{Gx^2 Gy^2} \tag{4}$$

The approximate value can be calculated according to the following relation:

$$|G| = |Gx| + |Gy| \tag{5}$$

The angle of the edge direction (for the pixel matrix) is given which leads to the slope as:

$$\phi = \text{artcn}\left(\frac{Gx}{Gy}\right)\text{-}3\pi 4 \tag{6}$$

Prewitt coefficient: This parameter is similar to Sobel eqaution and is used to detect vertical and horizontal edges in images equation as follow the mask of the prewitt coefficient of the vector x.y:

$$\begin{matrix} -1 & 0 & +1 \\ -1 & 0 & +0 \\ -1 & Gy & +1 \end{matrix} \qquad \begin{matrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{matrix}$$

Laplace-based edge detection techniques: This method depends on the trailing position of zero by the second derivative of the image to find the edges.

Laplacian of Gaussian (LoG): It was suggested by LOG[12]. 1982 Marr's image is the second derivative. The lattice coefficient highlights the areas of sudden change in intensity, so, they are often used to detect the edge. The lattice is applied to an image which is often performed with a softening filter through the Gaussian filter to reduce noise sensitivity. The parameter usually takes a gray scale image as input and the output is another gray scale image. The lattice (Y and X) L is given to an image which has the values of pixels (Y and X) l given with the following relationship:

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \tag{7}$$

Since, the input image is a series of discrete pixels, we must find a separate conversion nucleus through which the second derivative can be approached in the Laplace definition. Equation shows the cores used As follows equation cores used to approach the second derivation in the Laplace filter:

$$\begin{matrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{matrix} \qquad \begin{matrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{matrix}$$

Since, the nuclei converge with the second derivative on the image, they are very sensitive to the noise. To avoid this, apply Gaussian blur before applying the Laplace filter. As long as the conversion process is participative from Gauss to Laplace, it is possible to construct the transform of the Gaussian filter with the Laplace filter and then apply the resulting hybrid filter (Leslie Laplace of Gauss) to the desired result.

The application of the previous method offers two advantages: as long as the Laplace and Gauss nuclei are usually smaller than the image, this method requires relatively few computations. Also, the resulting hybrid nucleus[8] from Laplace after Gauss can be calculated in advance, so we need only one conversion to apply to the image in real time and thus short for processing time. To follow the LOG in binary space with the Gaussian standard deviation, follow the following format:

$$\text{Log}(x,y) = -\frac{1}{\pi\sigma^4} * \left[1\text{-}\left(\frac{x^2}{2\sigma^2} + \frac{y^2}{2\sigma^2}\right)e\text{-}^{(x^2+y^2)/2\partial^2}\right. \tag{8}$$

Equivalent matrix of LOG with standard deviation:

$$\begin{matrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{matrix}$$

Canny algorithm to detect the edges: Canny intentions were to provide something improved on the many edge detectors at the time. He has succeeded in achieving the goal and has presented in his study[13] his ideas and methods where he followed in his study a list of criteria to improve existing methods of edge detection.

The first clear standard was to reduce the error rate. It is important not to lose the edges of the image and not to respond to edges that do not exist. The second standard is to translate and define the edge points well. In other words, the difference between the edge pixels found by the edge explorer and the actual edge must be minimal.

The third standard is to create a single response to each edge. This standard was developed because the previous two standards were not sufficient to prevent a multiple response to the same edge. Based on these standard, the canny reagent of the edges softens the image first to remove[7], then the night of the image to highlight the spoken high-resolution image[8]. The algorithm then performs a trace through this speaker and removes each pixel that does not have an even greater value.

The slope matrix is reduced by hysteresis that uses it to track all remaining pixels that have not been removed. It uses two thresholds so that if the value for the pixel is less than the first threshold, zero (not an edge) is placed and if the value above the top threshold is an edge and if the value between the two thresholds is set 0 unless the pixel is in a path with another pixel above the threshold value 2 T In order to implement the Canny algorithm, you must follow these steps:

The first step is to filter any noise in the original image before attempting to identify and detect any edges. Because the Gaussian filter can be calculated using a simple mask, it is especially used in the Canny algorithm. Once the appropriate mask is calculated, Gaussian smoothing can be applied using the standard conversion method, and the conversion mask is usually much smaller than the actual image. As a result, the mask slides over the image, processing a square of pixels at a time. The larger the Gaussian mask, the less sensitive the detector is to noise. The error of positioning the detected edge increases somewhat as the Gauss mask is increased.

After smoothing and noise removal we perform the second step by finding the edge force by finding the slope of the image. The Sobelcoefficient[4] represents the night measurement system for image (2D).

In the third step, the direction of the edge is calculated by using the inclination in the Y and X directions. From which an error will be generated when the sum of X is equal to zero, s,o there should be restrictions in the code on this case. When the night is zero in direction X, in this case the direction of the edge must take the value 90° or 0°, depending on the value of the slope in the direction Y. If Gy is zero, the edge direction should equal zero, otherwise the edge value 90° The formula for finding the edge is given as follows:

$$\text{Theta} = \left(\text{Gy/Gx}\right) \tag{9}$$

Once you know the direction of the edge, the next step is to tie the direction of the edge toward the traceable image. So, if 5×5 pixels are distributed as follows images 55 pixels:

$$\begin{matrix} X & X & X & X & X \\ X & X & X & X & X \\ X & X & b & X & X \\ X & X & X & X & X \\ X & X & X & X & X \end{matrix}$$

By looking at pixel b we find four possible probabilities for directions when representing the surrounding pixels. Grade 0 (in horizontal direction), 45 (along the positive slash), 90 (vertical) or 135 (along the negative slash).

The edge orientation in one of these four directions should be resolved depending on which direction is closest (for example if the orientation angle is found to be equal to 3°, it is made equal to zero). If they are divided into five sectors, the angle of each sector belongs to the angle value of the four values mentioned above.

After you know the edge directions, you must follow the edge along the edge to remove each pixel that is not considered to belong to the edge (zero position). This process gives a fine line to the edge in the resulting image. Finally hysteresis[14] is used as a means of removing the resulting interruptions within the edge caused by the switching of the output of the coefficient between values above and below the threshold.

RESULTS AND DISCUSSION

The experiments were performed on two sets of color images which are samples of red blood cells containing parasites in their cells. Several criteria have been identified for assessing the resulting image quality as well as calculating the execution time and ambiguity of each edge detection algorithm. The target of edge discrimination is only to distinguish borders. Distinguish boundaries of importance in images such as red blood cell boundaries and parasite boundaries within them as accurately as possible.

Photo collections: Group A: contains 14 images of red blood cells containing parasites in their cell, a group characterized by high-luminance and low-contrast images. Figure 1 shows an example of the first set.

Group B: Contains 14 color images of red blood cells containing parasites in their cell, a group characterized by moderate illumination and relatively high variation. Figure 2 shows an example of the first set.

Image quality assessment criteria: The assessment is compared to the original image and gray scale image by viewing with the naked eye. The most important criteria for evaluation and comparison are: The accuracy of the edges of blood cells. This standard is based on the exact distinction of red blood cell edges. Identification of the limits of parasites: This standard is based on the precise distinction of parasite boundaries. Non-discrimination of abnormal boundaries.

Anomalies are the boundaries that appear in the resulting images and are not related to the objects in the original image but are caused by noise effects, reflection in the lighting, or the special case of the algorithm. Thus, this standard is dependent on avoiding the distinction of these anomalies continuity of edges.

This standard is based on the previous standards where the correct limits must be discovered and continuous with little interruptions and distortions Mystery (Entropy).

This standard was developed to give numerical comparison to the resulting images, since, the uncertainty is often used in the image classification stages[10]. It was observed that the algorithm with the best performance in terms of detected boundary accuracy is of great value to ambiguity compared with other algorithms.

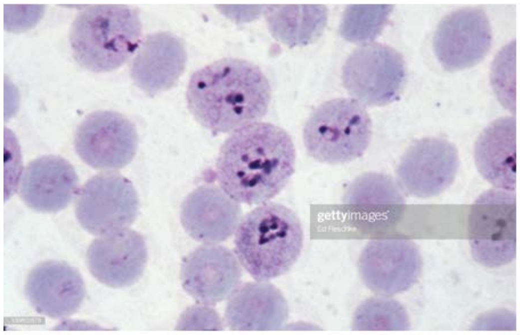

Fig. 1: Example of the first set of images

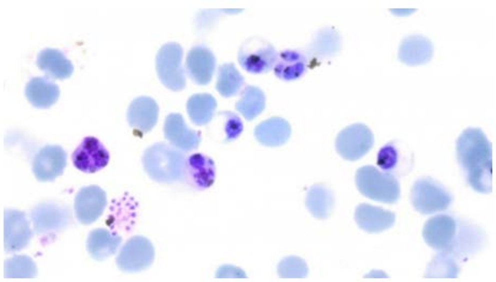

Fig. 2: Example of the second set of images

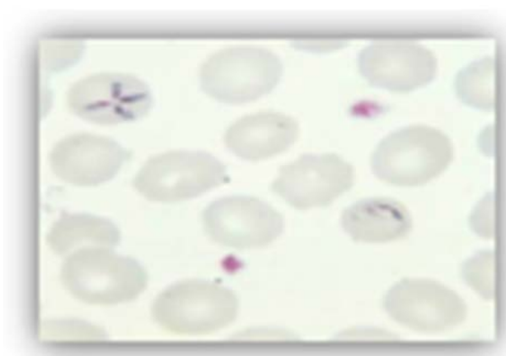

Fig. 3: Strong light group in RGB mode with the application of previuos techniques

Experimental work: The practical application was carried out and the results and comparisons were obtained in the MATLAB Student R2018a 9.2.0 and on the(HP-Notebook 15-ay108ne) with the Inter Core 15 7200U processor, 4GB DDR4 memory and 2GB of separate memory. We will display the sequence of action steps on the two sets of images and the results will be displayed for samples for each set of images mentioned previously.

The sequence of steps is as follows:

- We convert images from color to gray-scale in the low light and (strong and low light) sets because the previous edge detection algorithms work on gray scale images

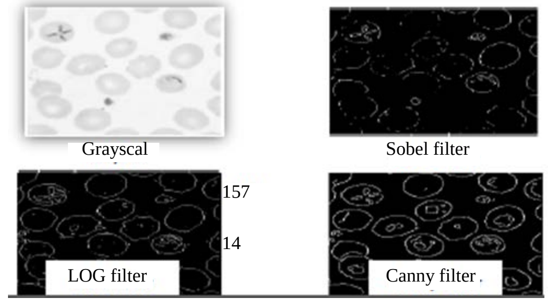

Fig. 4: The strong light group in RGB mode with previous techniques applied

- We then apply the detection algorithms to each image of each group and display the resulting images for the preview
- We take the uncertainty of each image in order to observe the values of the different images
- We compare the different results of the edge detection algorithms for each image and observe the results and record them

The Pseudo-code of the previous algorithm steps is as follows for each set of two images:

Input: List of Images
Output: Detected Edges
For each image in the list
Step1: Read an image
Step2: Convert the image from RGB to gray-scale
Step3: L1 = apply Sobel Algorithm
L2 = apply Robert Algorithm
L3 = apply Prewitt Algorithm
L4 = apply Laplacian of Gaussian Algorithm
L5 = apply Canny Algorithm
Step4: calculate Entropy for (L1, L2, L3, L4, L5) and Entropy forEach of them.
Step5: Show edges of (L1, L2, L3, L4, L5) and Entropy for each of them

The implementation of experiments on the first group appears as in Fig. 3. Perform experiments on the second group shown as in Fig. 4-6.

Results on group A: The implementation resulted in 14 images per algorithm to the ratios in Table 1 where the images resulting from the edge detection algorithms were classified to good accuracy, then medium and then weaker.

The average execution time for images by each edge detection algorithm is shown in Table 2 where we note that the largest implementation time is for the Roberts algorithm and the least time for the canny algorithm:

The average ambiguity of the set of images according to the algorithm of the detection of edges is shown in Table 3 where we note that the ratio of ambiguity is an algorithm and that the largest ambiguity is for the Canny and LOG algorithm and note that the ambiguity of the Canny and LOG algorithms are equal:

Table 1: Precision ratios of the techniques used in the first group

Algorithm	Good accuracy (%)	Medium accuracy (%)	Weak accuracy (%)
Sobel, Prewitt, Roberts	28.56	21.42	50.01
Laplacian of Gaussian (LOG)	42.88	50	7.11
Canny	28.56	28.56	42.85

Table 2: The time taken for each user on the first group

Algorithm	Sobel	Prewit	Robertst	LOG	Canny
Time average	0.251’s	0.262’s	0.317’s	0.231’s	0.195’s

Table 3: Ambiguity resulting from techniques used in the first group

Algorithm	Sobel	Prewit	Robertst	LOG	Canny
Time average	0.169	0.173	0.178	0.357	0.357

Table 4: The accuracy ratios of the techniques used in the second group

Algorithm	Good accuracy (%)	Medium accuracy (%)	Weak accuracy (%)
Sobel, Prewitt, Roberts	14.28	3571	50
Laplacian of Gaussian (LOG)	78.57	42.85	0
Canny	7.14	21.42	50

Table 5: The time taken for each technique used on the second group

Algorithm	Sobel	Prewitt	Roberts	LOG	Canny
Time average	0.224’s	0.220’s	0.228’s	0.215’s	0.189’s

Table 6: Ambiguity resulting from the techniques used in the second group

Algorithm	Sobel	Prewitt	Roberts	LOG	Canny
Ambiguity average	0.203	0.202	0.241	0.304	0.304

Table 7: Pros and cons of the techniques used

Coefficients	Props	Cons
Sobel, Prewitt, Roberts	Simplicity, detection of edges and directions	Sensitive to noise, poor accuracy
Laplacian of Gaussian (LOG)	Find the correct edge positions, test a wider area around the pixels	Limitations at angles and curves where gray-scale levels vary, you cannot find the edge orientation because of the Laplace filter
Canny	Finding precise edge positions, resistant to many types of noise	Poor performance in irregular lighting conditions, complex calculations

Results on the group B: The implementation resulted in 14 images per algorithm to the ratios in Table 4. The images resulting from the edge detection algorithms were classified as good, medium and weaker and in this group the accuracy was relatively close.

The average execution time for images by each edge detection algorithm is shown in Table 5 where the largest implementation time is for the Roberts algorithm and the least time for the Canny Algorithm:

The average ambiguity of the set of images according to each detection algorithm is shown in Table 6 where we note that the least ambiguity is for the algorithm and the largest ambiguity is for the Canny and LOG algorithm and note that the ambiguity of the Canny and LOG algorithms are equal:

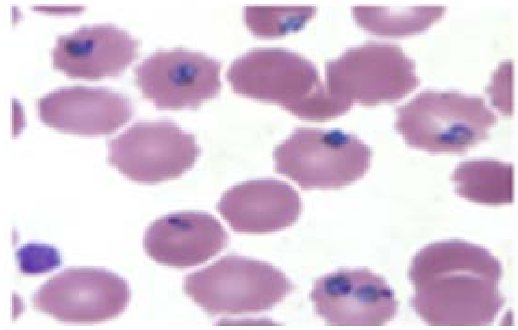

Fig. 5: Strong light group in RGB mode wih previous techniques applied

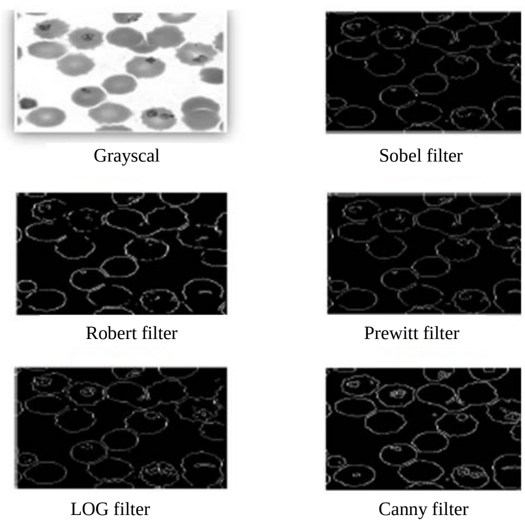

Fig. 6: Results of the pilot work

Pros and cons of the edge detection transactions used: By observing images, images and results from the use of previous transactions on the two sets of images, we can summarize the pros and cons of these transactions in Table 7.

CONCLUSION

The Laplacian of Gaussian (LOG) coefficient in the detection of edges gives the best accuracy on the first group, which is characterized by low contrast and high lighting by 42.88% and also the second group which is characterized by high contrast and moderate lighting at 78.57%. And that this algorithm gives better performance on images characterized by high contrast and moderate light. We note that the LOG and Canny algorithms give the highest average ambiguity of the resulting images by 0.357 on the first group and 0.304 on the second group. While we note that the Canny algorithm gives the best speed on the two groups of images by 0.195 on the first group and 0.189 on the second group and based on previous experiments, we recommend the use of LOG coefficient on medical cellular images that have characteristics similar to the images that we put in the experiments because of accuracy in the resulting with Knowledge that the Canny algorithm yields less accurate results than LOG but is a very acceptable accuracy. Through pre-processing on the lighting vehicle, we can increase the resulting accuracy Fig. 6 shows the results of the application of the LOG-based edge detection algorithm.

REFERENCES

01. Das, S., 2016. Comparison of various edge detection technique. Int. J. Signal Process. Image Pattern Recognit., 9: 143-158.
02. Shrivakshan, G.T. and C. Chandrasekar, 2012. A comparison of various edge detection techniques used in image processing. Int. J. Comput. Sci. Issues (IJCSI.), 9: 269-276.
03. Muthukrishnan, R. and M. Radha, 2011. Edge detection techniques for image segmentation. Int. J. Comput. Sci. Inf. Technol., 3: 259-267.
04. Kaur, S. and I. Singh, 2016. Comparison between edge detection techniques. Int. J. Comput. Appl., 145: 15-18.
05. Pirzada, S.J.H. and A. Siddiqui, 2013. Analysis of edge detection algorithms for feature extraction in satellite images. Proceedings of the 2013 IEEE International Conference on Space Science and Communication (IconSpace), July 1-3, 2013, IEEE, Melaka, Malaysia, pp: 238-242.
06. NagaRaju, C., S. NagaMani, R.G. Prasad and S. Sunitha, 2011. Morphological edge detection algorithm based on multi-structure elements of different directions. Int. J. Inf. Commun. Technol. Res., Vol. 1, No. 1.
07. Wang, C., T. Chen and Z. Qu, 2010. A novel improved median filter for salt-and-pepper noise from highly corrupted images. Proceedings of the 2010 3rd International Symposium on Systems and Control in Aeronautics and Astronautics, June 8-10, 2010, IEEE, Harbin, China, pp: 718-722.
08. Chan, R.H., C.W. Ho and M. Nikolova, 2005. Salt and pepper noise removal by median type noise detectors and detail preserving regularization. IEEE Trans. Image Proc., 14: 1479-1485.
09. Bin, L. and M.S. Yeganeh, 2012. Comparison for image edge detection algorithms. IOSR. J. Comput. Eng., 2: 01-04.
10. Wang, W., Y. Wang, Q. Huang and W. Gao, 2010. Measuring visual saliency by site entropy rate. Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 13-18, 2010, IEEE, San Francisco, USA., pp: 2368-2375.
11. Pujar, J.H., P.S. Gurjal, D.S. Shambhavi and K.S. Kunnur, 2010. Medical image segmentation based on vigorous smoothing and edge detection ideology. Int. J. Electr. Comput. Eng., 5: 121-127.
12. Wang, X., 2007. Laplacian operator-based edge detectors, pattern analysis and machine intelligence. IEEE Trans., 29: 886-890.
13. Yuan, S., D. Xiang, X. Liu, X. Zhou and P. Bing, 2017. Edge detection based on computational ghost imaging with structured illuminations. Optics Commun., 410: 350-355.
14. Nixon, M. and A. Aguado, 2019. Feature Extraction and Image Processing for Computer Vision. Academic Press, Amsterdam, Netherlands, Pages: 609.