1. Hiba-Allah T. Abdullah[1], 2. Riyadh Zaghlool Mahmood[2], 3. Sanabel M. Alhaj Zber[3], 4. Rasha A.Mohammed[4], 5. Marwa Riyadh Ahmed[5], 6. Azhar W. Talab[6]

Northern Technical University (1), *University of Mosul* (2), *Northern Technical University* (3), *Northern Technical University* (4), *Northern Technical University* (5), *Northern Technical University* (6)
ORCID: 1. 0000-0001-6276-0011; 2. 0000-0001-7899-6475; 3. 0000-0003-0232-9064; 4. 0000-0001-9346-5680; 5. 0009-0008-2195-0909; 6. 0000-0003-1429-3177

doi:10.15199/48.2024.02.05

FPGA-Based Three Edge Detection Algorithms (Sobel, Prewitt and Roberts) Implementation for Image Processing

***Abstract**- This technique aims to identify bone boundaries and fractures in noisy images by leveraging information from X-ray images. The computer-aided bone fracture detection method is primarily designed to help doctors generate improved diagnostic reports. Identifying accurate boundaries in noisy images remains challenging. Image processing algorithms have been limited to slow software implementations due to restricted processor speeds, necessitating a dedicated processor for edge detection. The Spartan3E-XC3S1600 FPGA kit will be employed to construct a fast architecture capable of performing edge detection using Sobel, Prewitt and Roberts edge detection systems.*

Streszczenie**- Ta technika ma na celu identyfikację granic kości i złamań na zaszumionych obrazach poprzez wykorzystanie informacji z obrazów rentgenowskich. Wspomagana komputerowo metoda wykrywania złamań kości ma przede wszystkim pomóc lekarzom w generowaniu lepszych raportów diagnostycznych. Identyfikacja dokładnych granic w hałaśliwych obrazach pozostaje wyzwaniem. Algorytmy przetwarzania obrazu zostały ograniczone do powolnych implementacji oprogramowania ze względu na ograniczoną prędkość procesora, co wymagało dedykowanego procesora do wykrywania krawędzi. Zestaw Spartan3E-XC3S1600 FPGA zostanie wykorzystany do zbudowania szybkiej architektury zdolnej do wykrywania krawędzi z wykorzystaniem systemów wykrywania krawędzi Sobel, Prewitt i Roberts.* (Implementacja algorytmów wykrywania trzech krawędzi opartych na FPGA (Sobel, Prewitt i Roberts) do przetwarzania obrazu***)

Keywords: MATLAB, FPGA, VGA port, LCD
Słowa kluczowe: MATLAB, FPGA, port VGA, LCD

1. Introduction

This section briefly addresses research, prior work, and demonstrations by various researchers who have attempted to elucidate the application of different edge detection filters, such as Sobel, Prewitt and Roberts, to software using MATLAB and hardware using various FPGAs related to those used in this paper.

In 2011, Mahendran and Santhosh introduced an automated system to detect long bones, specifically tibia bones, by employing a fusion-classification approach. The process consisted of four stages: preprocessing, segmentation, feature extraction, and bone detection, and utilized various image processing techniques for effective fracture identification. The fusion classification incorporated three classifiers: BPNN, SVM, and NB, resulting in notable enhancements in detection rate and classification speed [1].

In 2012, Li Bin and Mehdi Samiei Yeganeh examined and compared multiple traditional edge detection algorithms, such as Roberts, Sobel, Prewitt, LOG, and Canny, using MATLAB. They found that the Sobel operator was better at detecting diagonal edges, while the Prewitt operator excelled at horizontal and vertical edges. The LOG generated double-width edges relative to pixels, and the Canny algorithm effectively identified weak edges due to its noise resistance [2].

In 2013, Dhanabal and his team created a co-processor for image processing as part of a project centered on image processing. This co-processor was designed specifically for edge detection, and the algorithm was implemented on an FPGA to take advantage of inherent parallelism, resulting in enhanced performance. The architecture was similar to an ARM processor, which served as the master and held images for processing. Images were sent from the ARM to the FPGA for processing and were displayed through a VGA. The edge detection core was implemented within the FPGA, where the image was read, processed, and saved in memory before being shown by the VGA controller. The Sobel Edge detection algorithm was employed, demonstrating efficiency for smooth edges and reduced sensitivity to noise [3].

In 2014, Vaishnav Tej Akhil, Prof. Amit Kumar, and Prof. Ekta Chotai proposed a comprehensive method for implementing an edge detection algorithm on FPGA using VHDL. The results showed that the Sobel algorithm performed best, and they also implemented software-based algorithms on hardware using Spartan/Altera kits [4].

In 2015, Mohammad and Brendan presented a high-speed pipeline-based FPGA architecture for first-order derivative edge detection techniques. This architecture used parallelism at various levels to accelerate operations. They suggested one-way and two-way parallel architectures, employing Verilog HDL for Cyclone IV FPGA with different edge detection methods. The results showed speed-ups of 460 and 920 for the parallel architectures, respectively [5].

Also in 2015, High-Level Synthesis (HLS) technology emerged, enabling the automatic transformation of high-level language programs into hardware language programs. Kazunari Yosikawa, Naohiro Iwanaga, Hamachi Tatsuya, and Akira Yamawakim created a Canny edge filter using fixed-point number HLS. They experimentally compared hardware modules utilizing fixed-point numbers with those using floating-point numbers, discovering that fixed-point programs were transformed by HLS into more efficient, compact hardware modules with increased operational speeds. The researchers exhibited a 76% reduction in hardware size and a 33% decrease in execution time, leading to a highly effective fixed-point Canny edge filter hardware capable of HLS conversion [6].

In 2017, Avinash G. Mahalle and A. M. Shah implemented classical edge detector operators, including Robert, Prewitt, Sobel, and Canny edge detection algorithms on a Spartan-3E FPGA. The results indicated that Canny was the best edge detection algorithm, followed by Sobel, Prewitt, and Roberts. Based on application needs, they could use any of these algorithms in real-time. They also discovered that the JTAG hardware co-simulation approach using Xilinx System Generator was a straightforward and efficient method for implementing hardware on an FPGA platform [7].

In 2018, Wint and colleagues concentrated on identifying and classifying fractured or non-fractured tibia bones in the lower leg using X-ray images. The fracture detection process for tibia bones involved three primary steps: preprocessing, feature extraction, and classification and localization of fracture sites. During preprocessing, Unsharp Masking (USM), an image sharpening technique, was employed to enhance images and highlight edges. The system was implemented using MATLAB, a programming software equipped with an extensive range of image processing tools. The fracture classification system attained an accuracy of 82% [8].

2. Edge detection techniquwes

Most edge detection methods are based on the concept of 2D spatial filtering operations, which involve continuous alterations in pixel values. In contrast to other edge detection techniques, first difference-based operator algorithms are the easiest to implement on hardware. In the Sobel Algorithm, gradient computation relies on both horizontal and vertical masks at each image point. The convolution, when compared to the image, is relatively small. Consequently, starting from the top left corner of the image, horizontal and vertical masks slide across the entire image. Moreover, the masks move rightward on the image until they reach the end of the row. Then, they resume from the leftmost part of the image's second row, as illustrated in Figure 1 [9].

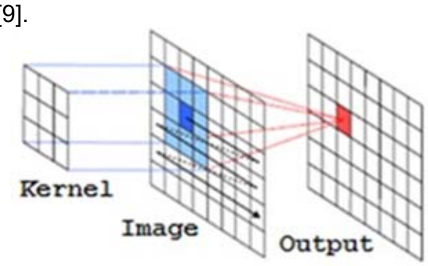


Fig **1**: 2D spatial filtering operation

A.Sobel Edge Detector

The Sobel operator, a well-known first-order edge detection operator, computes the approximation of image intensity. At every point in the image, the Sobel operator produces the associated norm of the gradient vector. To calculate the image approximation, it uses two 3x3 masks convolved with the original image. Both convolution masks aim to identify edges in both horizontal and vertical directions. The Gx and Gy masks are determined using the following equations [9][10].

(1) $$G_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$ (2) $$G_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

B.Prewitt Edge Detector

The Prewitt operator's output determines the norm of the gradient vector at every point in the image. This is achieved by convolving the original image with a small, separable, integer-valued filter in the horizontal direction using equation 3, vertical direction using equation 4, and both diagonal directions using equations 5 and 6. It is also a first-order derivative operator [11]. The approach is based on the following:

(3) $$A = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$ (4) $$B = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

(5) $$C = \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix}$$ (6) $$D = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix}$$

C. Robert Edge Detector

This operator is gradient-based. It begins with the calculation of the sum of the squares of the differences between diagonally adjacent image pixels using discrete differentiation. Subsequently, it computes the approximate gradient of an image. Equations (7) and (8) demonstrate the convolution of the input image with the operator's default kernels, followed by the calculation of magnitude and directions using 2x2 kernels [12]. Figure 2.6 illustrates the image and its edge detection using this method:

(7) $$E = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$ (8) $$F = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

3. Hardware Computer Aided Fracture Detection System

In Figure 2, the aim of the Signal Analysis segment is to identify bone fractures in X-ray input images using a variety of edge detection methods. In the second setup, two architectures are devised: one for detecting Sobel, Prewitt, and Roberts filters. In the following sections, comprehensive descriptions of each part will be given. Keep your text and graphic files separate until the text is formatted and styled. Refrain from using hard tabs and restrict hard returns to only one at the end of a paragraph. Do not include any page numbers within the paper. Also, avoid numbering text headings, as the template will handle that for you.

Fig.2: The Steps of the Designed Fracture Detector System

4. Architecture of Three filters

Figure 3 displays the block diagram of the first architecture used for detecting bone fractures. This architecture includes two switches for selecting the operation mode, an integrated rotary encoder for managing the threshold value of each edge detection filter, and two BRAMs; the first one for input image data and the second one for storing output results. Defining abbreviations and acronyms is not necessary. Refrain from using abbreviations in the title or headings unless it is absolutely essential.

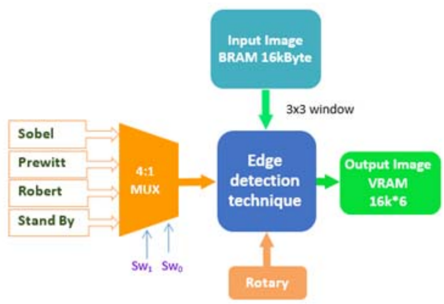


Fig 3: Sobel, Prewitt, and Robert Block

Figure 4 illustrates the flow chart of the steps followed in the design of Sobel, Prewitt, and Roberts filters by using the designed system.

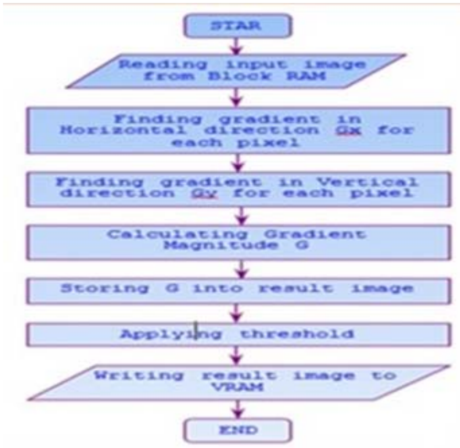
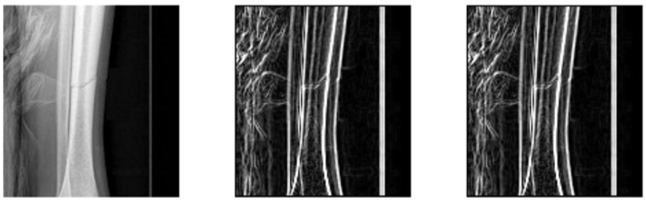


Fig. 4: The Flowchart of Sobel, Prewitt, and Roberts filters implementation process

a. **Filters Edge Detector Performance**

Upon reading the fractured bone's colored image file using MATLAB with an image size of 128x128 pixels, the image is converted to gray scale using a specific programming statement before applying the Sobel filter.
Figures 5 show the results obtained using MATLAB, with images featuring 8-bit pixel resolution in (a) and 6-bit pixel resolution in (b).



Original (a) 8-bit Result (b) 6-bit Result

Fig. 5: Simple fracture image MATLAB results using Sobel filter

Figure 6 presents the hardware outcomes from the chip being tested. The result image without a threshold (a) and with a suitable threshold (b) is shown on an external screen, while figures 7 illustrate the appropriate threshold choices for the Simple and Communicated images, respectively.

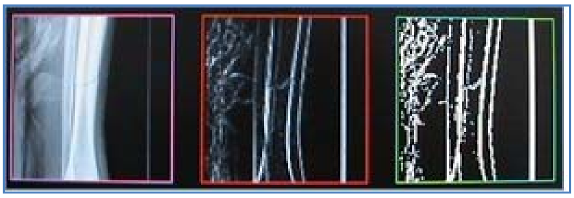
Original image (a) without threshold (b)with threshold

Fig. 6: Simple fracture image FPGA results using Sobel filter

Fig. 7: Threshold selection for the Communicated fracture image using Sobel filter

The Prewitt edge detection filter was also executed using MATLAB and VHDL, taking into account the calculation of execution time. The same two images employed in the earlier detector were selected to display the results of the implementation using MATLAB and VHDL for comparative purposes.

Figures 8 present the hardware outcomes from the chip being tested. The result image without a threshold (a) and with a suitable threshold (b) are shown on an external screen, while figures 9 illustrate the appropriate threshold choices for the Simple and Communicated images, respectively.



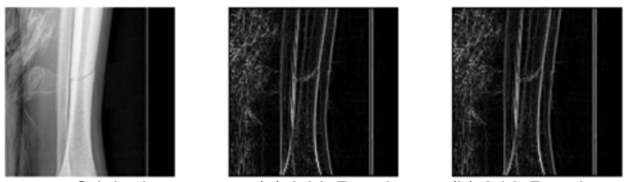
Original image (a) without threshold (b) with threshold

Fig. 8: Simple fracture image FPGA results using Prewitt filter

Fig 9: Threshold selection for Simple fracture image using Prewitt filter

The Roberts edge detection filter was also implemented using MATLAB and VHDL, and the same two images used in the previous detector were selected to showcase the results of the implementation using MATLAB and VHDL for comparison purposes.

As demonstrated in the earlier detectors, after reading the colored image file of the fractured bone using MATLAB with an image size of 128x128 pixels, the image is converted to gray scale using a specific programming statement before applying the Roberts filter. Figures 10 display the results obtained using MATLAB, with images featuring 8-bit pixel resolution in (a) and 6-bit pixel resolution in (b).



Original (a) 8-bit Result (b) 6-bit Result

Fig. 10: Simple fracture image MATLAB results using Roberts filter

Original image (a) without threshold (b) with threshold

Fig 11: Simple fracture image FPGA results using Roberts filter

Fig. 12: Threshold selection for Simple fracture image using Roberts filter

Figures 11 showcase the hardware outcomes acquired from the chip under examination. The result image without a threshold (a) and with an appropriate threshold (b) are displayed on an external screen, while Figures 12 highlight the suitable threshold choices for the Simple and Communicated images, respectively.

b. Thresholding Variation Performances

Thresholding is a simpler method for edge detection in images. It can be used to generate binary images from a grayscale image. In this technique, each grayscale pixel in an image is replaced with either a black or a white pixel, depending on the pixel intensity. If the image pixel intensity is below a specific threshold value, the result will be a black pixel, whereas if the image pixel intensity is above the threshold, the outcome will be a white pixel. This applies to all previously filtered images, as demonstrated in Table 1.

Table 1: Images threshold table

	Image name	Sobel	Priwett	Roberts
1.	Simple	5DH (93)	36H (54)	21H (33)
2.	Communicated	5FH (95)	52H (82)	29H (41)
3.	Samp_1.JPG	39H (57)	2DH (45)	11H (17)
5.	Samp_2.JPG	43H (67)	2EH (46)	1AH (26)
4.	Samp_3.JPG	2DH (45)	21H (33)	0DH (13)

Figure 13 provides the UCF constraints for the VGA display port, including the I/O pin assignment, the I/O standard used, the output slew rate, and the output drive current.

```
NET "VGA_RED"   LOC = "H14" | IOSTANDARD = LVTTL | DRIVE =
                8 | SLEW = FAST ;
NET "VGA_GREEN" LOC = "H15" | IOSTANDARD = LVTTL | DRIVE =
                8 | SLEW = FAST ;
NET "VGA_BLUE"  LOC = "G15" | IOSTANDARD = LVTTL | DRIVE =
                8 | SLEW = FAST ;
NET "VGA_HSYNC" LOC = "F15" | IOSTANDARD = LVTTL | DRIVE =
                8 | SLEW = FAST ;
NET "VGA_VSYNC" LOC = "F14" | IOSTANDARD = LVTTL | DRIVE =
                8 | SLEW = FAST ;
```

Fig 13: UCF Constraints for VGA Display Port

Figure14 provides the UCF constraints for the Character LCD, including the I/O pin assignment and the I/O standard used.

```
NET "LCD_E" LOC = "M18" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW =
SLOW ;
NET "LCD_RS" LOC = "L18" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW =
SLOW ;
NET "LCD_RW" LOC = "L17" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW =
SLOW ;
# The LCD four-bit data interface is shared with the Strata Flash.
NET "SF_D<8>" LOC = "R15" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW =
SLOW ;
NET "SF_D<9>" LOC = "R16" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW =
SLOW ;
NET "SF_D<10>" LOC = "P17" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW =
SLOW ;
NET "SF_D<11>" LOC = "M15" | IOSTANDARD = LVCMOS33 | DRIVE = 4 | SLEW =
SLOW ;
```

Fig 14: UCF Location Constraints for the Character LCD

5. Conclusions

- It has been found that the Roberts algorithm for edge detection is more suitable for this kind of application.
- The outcomes of edge detection algorithms differ based on their filters. It is observed that the Sobel algorithm is the most effective for this purpose compared to others.
- The three edge detection algorithms are implemented using the Spartan 3E-XC3S500, which has sufficient resources for the aforementioned filters.

Authors

Mrs. Hiba-Alla Tariq Abdullah *is an engineer from Department of Computer Technology Engineering, she received bachelors from Northern Technical University 2001 / Engineering Technical College-Mosul-Iraq / Department of Computer Technology Engineering. She received Master degree from Northern Technical University at 2020 in computer technology engineering / Engineering Technical College-Mosul-Iraq.*
Email: hibatallahtariq@ntu.edu.iq.

Mr. Riyadh Zaghlool Mahmood *is a lecturer in Cyber Security Department, College of Computer and Mathematic Science, University of Mosul; He received his Bsc., Msc. and Ph.D. degrees in Computer Science from University of Mosul in 1991, 2005 and 2012 respectively.E-Mail: riyadh.zaghlool@uomosul.edu.iq*

Mrs. Sanabel M. Alhaj Zber *born in Mosul, Iraq in 1973, She received the M.Sc. in electrical engineering from university of Mosul, Iraq, in 2001. Then she joined the Technical college of Mosul, Northern Technical University, Mosul, Iraq, as a lecturer. She has publications in many conferences and journals. Here works focus on the renewable energies systems. Her areas of interests are Renewable energy, Hybrid power systems, and A.C. Machine.E-Mail: Sanabel.m.mohammed@ntu.edu.iq*

Mrs. .Alya H. AL-Rifaie*: is a Asst. Prof. at Dep. of techniques/Engineering Technical College /Mosul , and received the M.Sc. degrees in electrical engineering from university of Mosul, Iraq, in 2008. Then she joined to the technical college of Mosul, Northern Technical University, Mosul, Iraq. She has many publications in conferences and scientific journals on renewable energies systems. Her areas of interests are Renewable energy, Hybrid power systems, HVDC, Protection and A.C. Machine. E-Mail :* alya.hamid@ntu.edu.iq

Mrs. Marwa Riyadh Ahmed *obtained her B.Sc. (2007) and M.Sc. (2019) in Computer Engineering Technology from Northern Technical University (NTU). Currently, she is working as assistant Lecturer in Computer Engineering department, in Northern Technical University (NTU)/ Technical College of Engineering, Mosul-Iraq. E-Mail :* marwa.riyadh@ntu.edu.iq

Mrs. *Azhar Waleed Talab obtained her B.Sc. (2007) and M.Sc. (2022) in Computer Engineering Technology from Northern Technical University (NTU). Currently, she is working as assistant lecture in Computer Engineering department, in Northern Technical University (NTU)/ Technical College of Engineering, Mosul-Iraq. E-Mail :* Azhar.w@ntu.edu.iq

REFERENCES

1. S.K.Mahendran, S.Santhosh Baboo, 2011, "An Enhanced Tibia Fracture Detection Tool Using Image Processing and Classification Fusion Techniques in X-Ray Images", Volume XI Issue XIV Version I, Volume 11 Issue 14 Version 1.0 August 2011
2. Li Bin, Mehdi Samiei yeganeh, 2012, "Comparison for Image Edge Detection Algorithms", IOSR Journal of Computer Engineering (IOSRJCE), ISSN: 2278-0661 Volume 2, Issue 6 (July-Aug. 2012), PP 01-04.
3 Vaishnav Tej Akhil, Prof.Amit Kumar, Prof.Ekta Chotai, 2014, "Hardware Implementation of Edge Detection Algorithms", International Journal Of Engineering Development And Research | IJEDR(www.ijedr.org) (Two Day National Conference (RTEECE-2014) -17th ,18th January 2014)
4 Mohammad Shokrolah Shirazi, Brendan Tran Morris, 2015, "Fast Edge Detection Architecture using Different Levels of Parallelism on a FPGA", International Journal of Computer Applications (0975 8887), Volume 113 - No. 13, March 2015
5 Kazunari Yosikawa, Naohiro Iwanaga, Hamachi Tatsuya, Akira Yamawaki , 2015, "Development of Fixed-point Canny Edge

Filter", Operation for High-level Synthesis", The Institute of Industrial Applications Engineers, Japan.
6 Avinash G. Mahalle, A. M. Shah, 2017, "FPGA Implementation of Gradient Based Edge Detection Algorithms", Copyright to IJIRCCE DOI: 10.15680/IJIRCCE.2017. 0505087, Vol. 5, Issue 5, May 2017.
7 Wint Wah Myint, Khin Sandar Tun, Hla Myo Tun, 2018, "Analysis on Leg Bone Fracture Detection and Classification Using X-ray Images", Wint Wah Myint, Khin Sandar Tun, Hla Myo Tun. Analysis on Leg Bone Fracture Detection and Classification Using X-ray Images. Machine Learning Research. Vol. 3, No. 3, 2018, pp. 49-59. doi: 10.11648/j.mlr.20180303.11
8 Umang I. Patel, Hina Patel, 2014, "A Review on Edge Detection Techniques Based on FPGA", e-ISSN: 2349-9745; p-ISSN: 2393-8161, International Journal of Modern Trends in Engineering and Research, IJMTER-2014, All rights Reserved
9 Pooja Sharma,Gurpreet Singh ,Amandeep Kaur, 2013, "Different Techniques Of Edge Detection In Digital Image Processing", Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 3, Issue 3, May-Jun 2013, pp.458-461
10 Sheetal D. Bhoyar(ME, BE) , Tarnnum Pathan (Mtech , BE), Amit D. Landge(Mtech , BE), 2016, "Design and Implementation of Sobel Edge Detection technique using VHDL", International Journal of Advanced Research in Science, Engineering and Technology Vol. 3, Issue 5 , May 2016
11 Abdulbasit Alazzawi, Husam Alsaadi, Abidaoun Shallal, Saad Albwi, 2015, " Edge Detection-Application Of (First And Second) Order Derivative In Image Processing", Diyala Journal of Engineering Sciences, ISSN 1999-8716, Second Engineering Scientific Conference College of Engineering –University of Diyala 16-17 December. 2015, pp. 430-440
12 Satbir Kaur, Ishpreet Singh, 2016, "Comparison between Edge Detection Techniques", International Journal of Computer Applications Volume 145 – No.15, July 2016