

Implementation of Image Processing Techniques using LabVIEW and MATLAB

B. Janardhana Rao

Assoc. Professor, CVR College of Engineering/ECE Department, Hyderabad, India

Email: janardhan.bitra@gmail.com

Abstract: Image processing play a major role in day to day life. The implementation of image processing applications using MATLAB takes place in practice. The block based implementation of image processing applications can be done with LabVIEW (Laboratory Virtual Instrument Engineering Workbench). This paper discusses the basic operations on images like extraction of the RGB (Red, Green, and Blue) components in a color image, converting the gray scale image to binary image and the edge detection process in both MATLAB and LabVIEW. Based on the experimental results, the merits and demerits of programming languages are discussed.

Index Terms: MATLAB, LabVIEW, RGB and edge detection.

I. INTRODUCTION

LabVIEW is a visual programming language and well suits for hardware integration. LabVIEW code is much easier to debug and as a visual programming language it contains less number of bugs in the code. Most of the times LabVIEW runs on hardware than normal computer. LabVIEW will be used for multidisciplinary applications. MATLAB is text based programming language and it suits well for matrix algebra and data visualization. In general MATLAB is the best tool for image processing applications. In this paper, the basic image processing operations are implemented using LabVIEW and MATLAB. An image is a two dimensional matrix of pixel values. There are three types of images in general, hence three different types of pixel matrices. The black and white image (binary image) consists of pixel values as ones and zeros, ones indicate the brighter portions and zeros indicate the darker portions [1]. The gray scale image consists of pixel values as 0 to 255. Like gray scale image color image is also a matrix of pixels values from 0 to 255. Color image is also called as RGB image. RGB image is an overlap of matrices for red, green and blue components of size 256x256. Then the total image size becomes 256x256x3. Since the image is represented as matrix, any mathematical operations on images can be performed on image that can be done with a matrix [2].

166	167	167	169	168	167	166	168	167	167
167	167	168	170	169	167	166	167	167	165
168	168	168	169	169	167	166	170	166	166
168	168	169	169	169	168	167	171	166	166
170	169	168	169	169	168	167	171	168	167
172	169	167	167	169	169	168	172	168	167
171	169	168	167	169	169	168	171	170	167
172	172	171	168	169	169	169	173	167	164
167	167	169	167	168	169	171	168	168	169

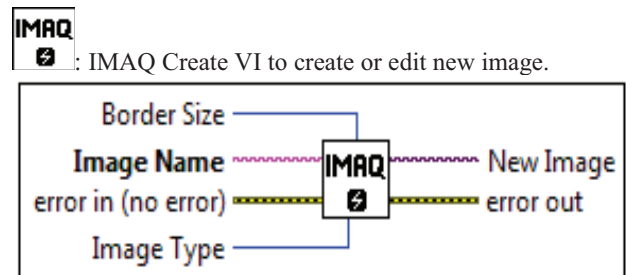
Figure 1. Gray scale image matrix

The edge detection techniques are applicable only for black and white images, so in order to apply the edge detection to RGB or gray scale image first it should be converted into binary image. As it is not also possible to convert RGB images to black and white directly, first it is to be converted into gray scale then binary image.

There are different types of edge detection techniques which are Roberts, Prewitt, Sobel and Canny. In this paper canny edge detection is implemented.

II. LABVIEW PROGRAMMING

LabVIEW is a graphical programming environment by National Instruments (NI). LabVIEW program consists of two major components: Front Panel (FP) and Block Diagram (BD). A front panel provides a graphical user interface and block diagram contains building blocks of a system resembling a flowchart [3], few icons used in this paper are given below [4]:



Reads an image file, including any extra vision information saved with the image.

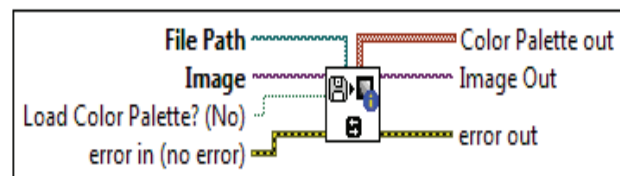


Image Type : Image Type to choose types of image (Gray Scale (U8), Gray Scale (I16), RGB (U32), HSL (U32)).

File Path : File Path to choose the path of image (.png) only



: Image information indicator to display the image

III. IMPLEMENTATION USING MATLAB

The MATLAB (Matrix Laboratory) is a proprietary programming language developed by MathWorks. It contains thousands of built in functions for wide variety of concepts in different specializations. It includes more number of Toolboxes for solving the problems in various fields of research like Signal Processing, Image Processing, Communications, Fuzzy logic, Neural Networks etc. It contains the Graphical User Interface (GUI) for better understanding the solution of various problems [5].

In this paper MATLAB programming is implemented for basic operations on images using Image Processing toolbox which are extraction of RGB components from the RGB color image and canny edge detection method. These image processing concepts are implemented here for comparing the results with LabVIEW programming.

In RGB color image, the colors appear as spectral components of red, green and blue. The color of the pixels in RGB image is formed by the combination of red, green and blue components, these are called as color channels or color components [6]. The intensity values of the RGB color image is represented as

$$I_{RGB} = (I_R(x, y), I_G(x, y), I_B(x, y))$$

Where $I_R(x, y)$ is intensity of the pixel at (x, y) in the red channel, $I_G(x, y)$ is the intensity in the green channel and $I_B(x, y)$ is the intensity in the blue channel.

Figure 3, 4 and 5 shows the red, green and blue components of the input image shown in figure 2.

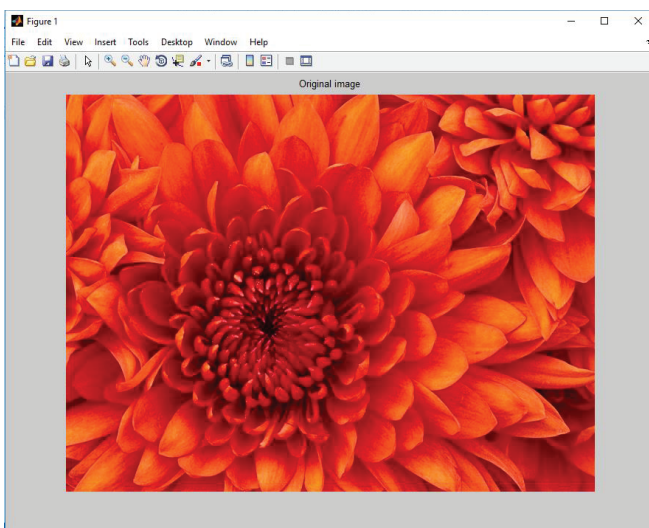


Figure 2. Input image

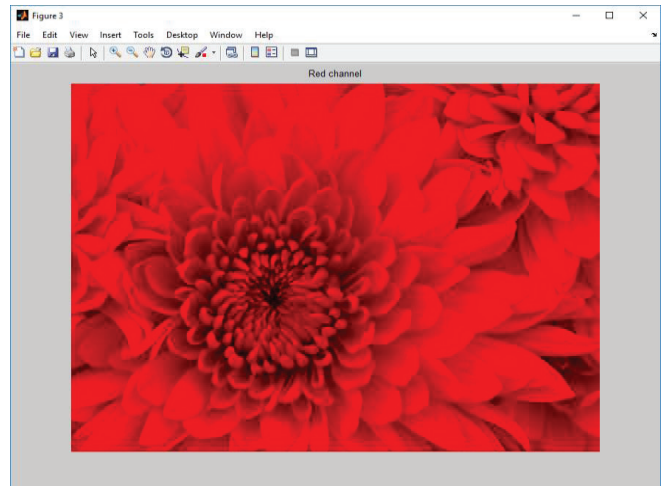


Figure 3. Red Component in RGB color image

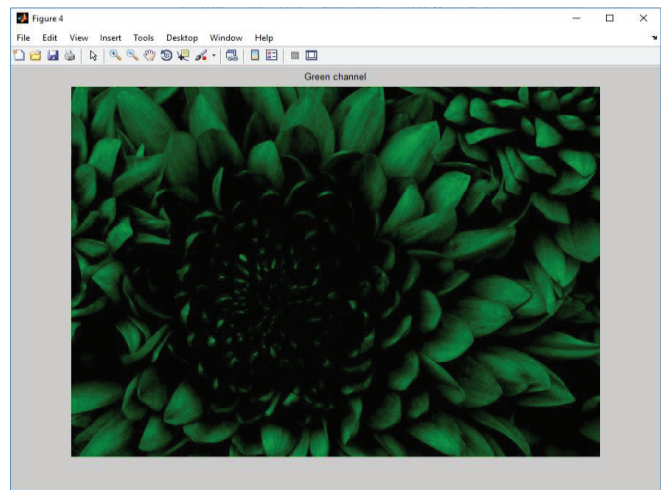


Figure 4. Green Component in RGB color image

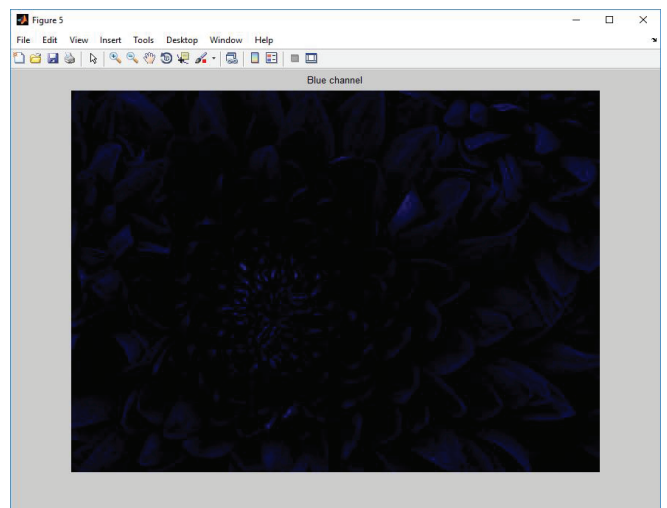


Figure 5. Blue Component in RGB color image

The edge detection of the images are used for many applications in the image processing [7]. The edge detection of the images is carried out by the flowchart shown in figure 6.

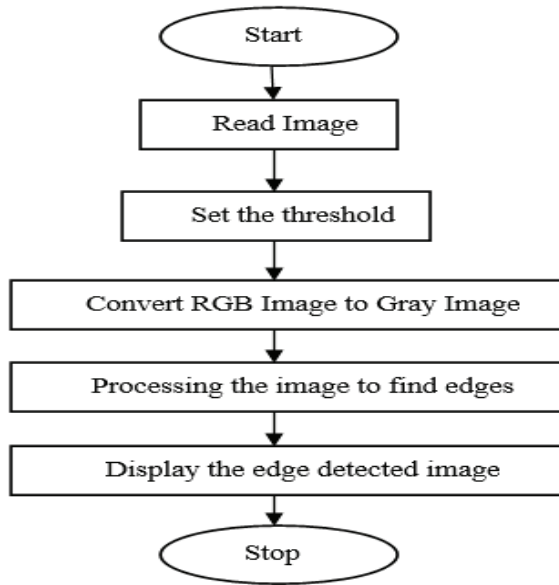


Figure 6. Flow chart for edge detection

There are different types of edge detection techniques and out of these canny edge detection technique gives the good results. Figure 8 shows the output image obtained by applying the canny edge detection technique to the input image shown in figure 7.

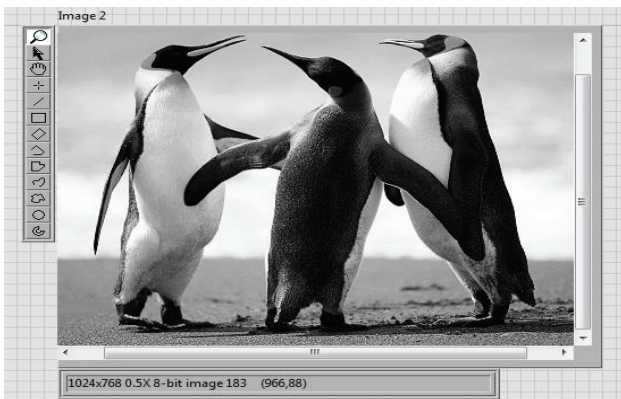


Figure 7. Input gray scale image

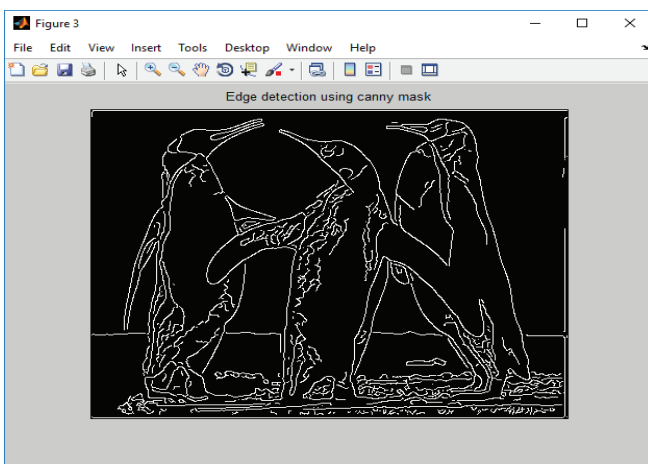


Figure 8. Output image from canny edge detection using MATLAB

IV. IMPLEMENTATION USING LABVIEW

The Front panel view of LabVIEW is shown in figure 9. The applications of Digital Image Processing are implemented using LabVIEW toolkit 2015.

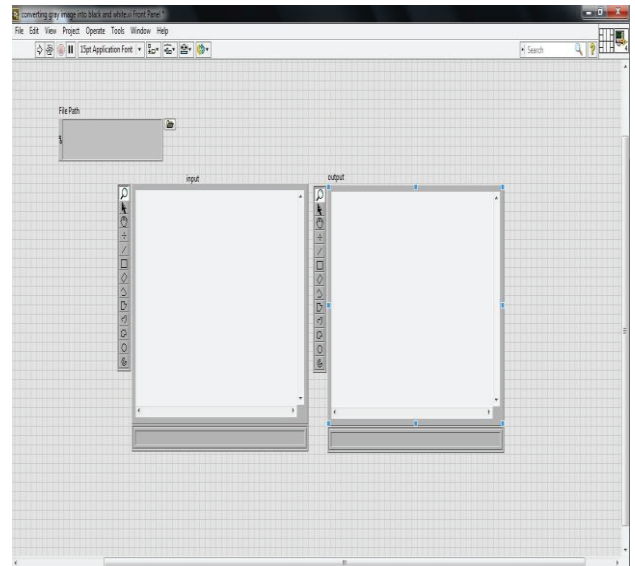


Figure 9. Front Panel of LabVIEW

The block diagram for extracting RGB components from the color image is shown in figure 10. Figure 11, 12 and 13 shows the red, green and blue components respectively obtained for the input image shown in figure 2.

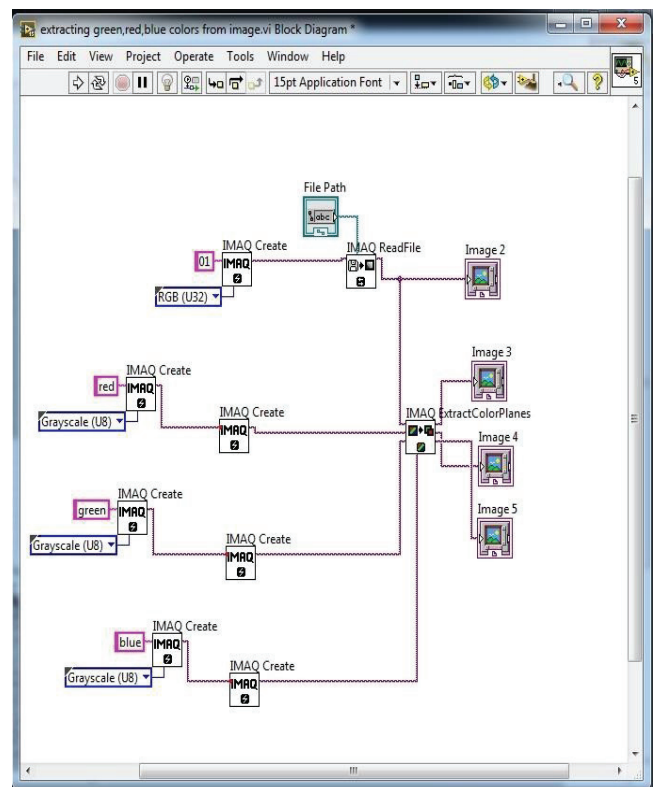


Figure 10. Block diagram for extracting RGB components

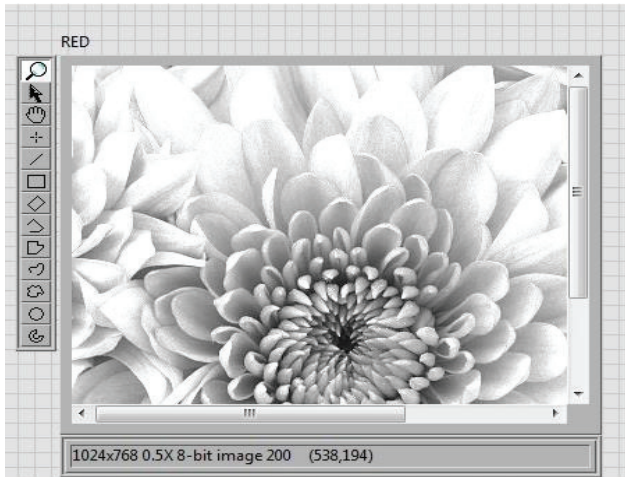


Figure 11. Red Component in color image

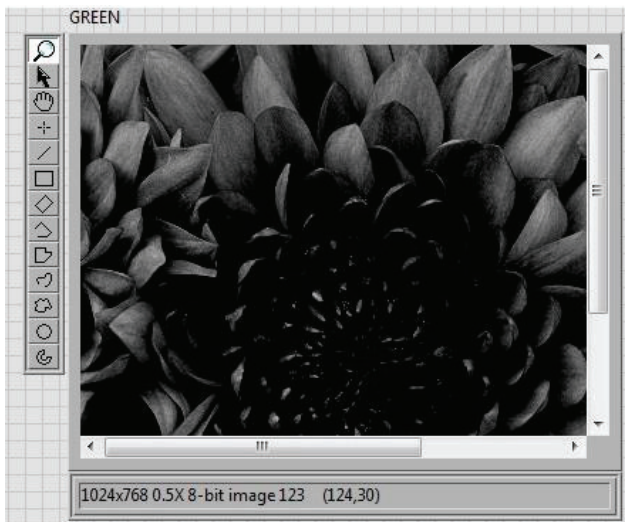


Figure 12. Green Component in color image

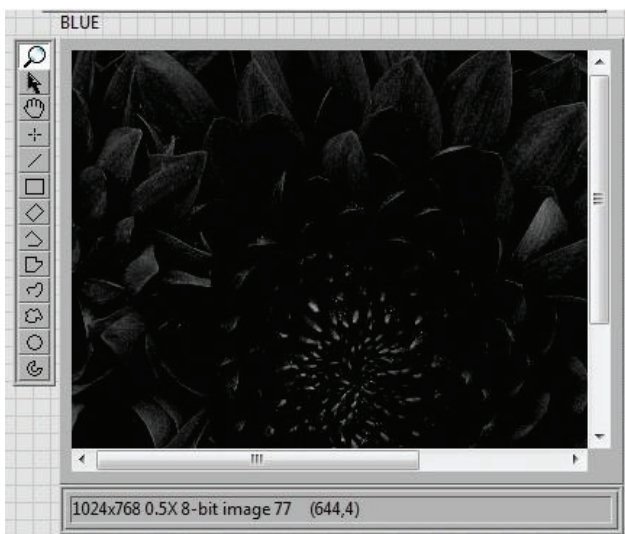


Figure 13. Blue Component in color image

The block diagram for converting gray scale image to binary image is shown in figure 14.

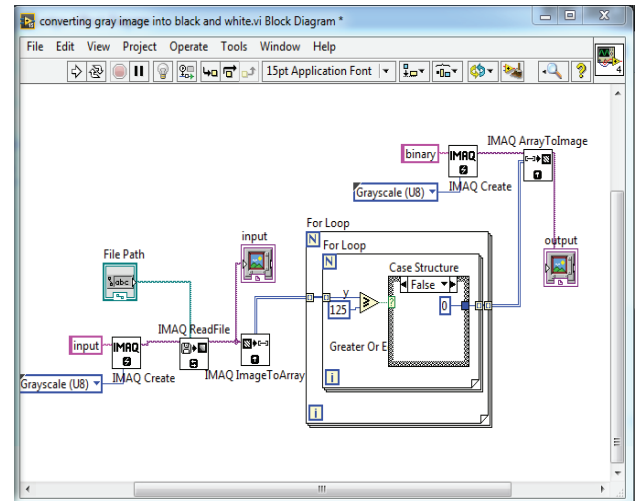


Figure 14. Block diagram for converting gray scale image into binary image

By applying the image shown in figure 2 as input to the block diagram shown in figure 14 to convert the gray scale image into binary image, the output image obtained is shown in figure 15.

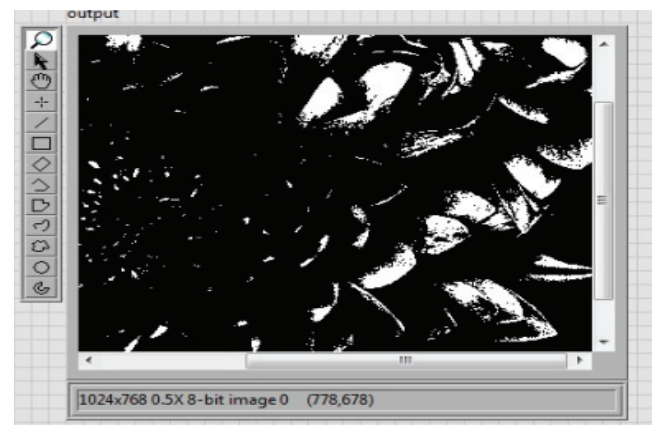


Figure 15. Binary image from grayscale image

The block diagram for implementation of edge detection for color image using canny edge detection method is shown in figure 16.

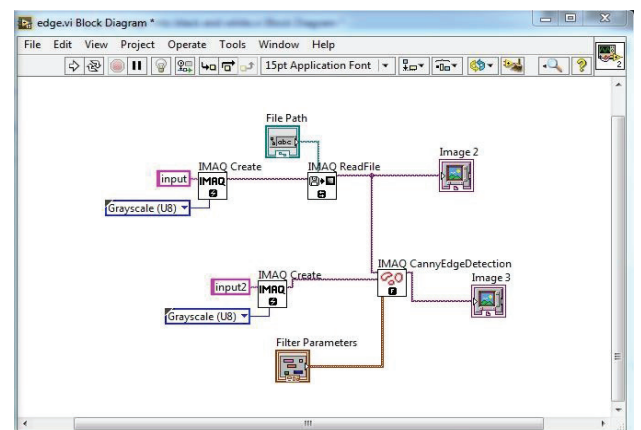


Figure 16. Block diagram for canny edge detection

The input image applied to the canny edge detection method implemented in figure 16 is shown in figure 7. The output image obtained from the canny edge detection method [8, 9] implemented in LabVIEW is shown in figure 17.

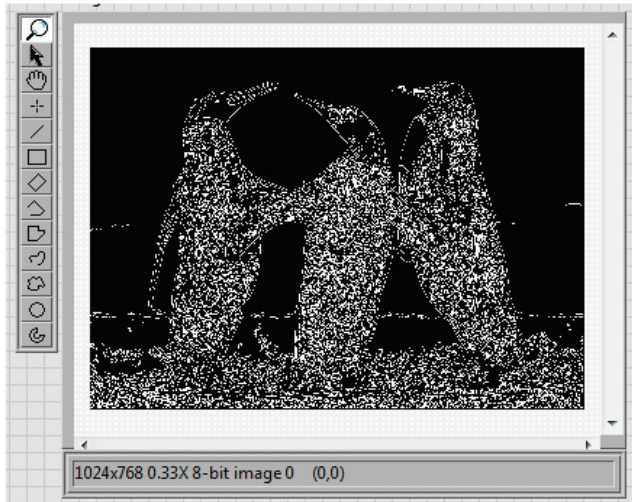


Figure 17. Output image from canny edge detection using LabVIEW

V. CONCLUSIONS

In this paper the basic image processing applications are implemented using MATLAB and LabVIEW. The extraction of RGB components from RGB color image and canny edge detection technique is implemented using both MATLAB and LabVIEW programming. The conversion of gray scale image to binary image implemented using LabVIEW is also presented. From the experimental results it is concluded that the implementation of image processing applications is easy and more efficient in LabVIEW compared to MATLAB in real time applications. MATLAB programming takes more processing time compared to LABVIEW programming.

REFERENCES

- [1] Rafael C. Gonzalez (University of Tennessee), Richard E. Woods (MedData Interactive) and Steven L. Eddins (The MathWorks, Inc.), in '*Digital Image Processing Using MATLAB*' Second Edition, 2009 by Gatesmark, LLC.
- [2] S.Sridhar, Digital Image Processing, Oxford University Press, 2011.
- [3] N. Kim, N. Kehtarnavaz, and M. Torlak," LABVIEW Based Software Defined Radio: 4-QAM Modem", Journal of Systemic, Cybernetics and Informatics (IISCI), Vol. 4, No.3, ISSN 1690-4524, 2006.
- [4] G. C. Panayi, —Implementation of Digital Image Processing Functions Using LabVIEW, M.S. thesis, Dept. of Electrical and Computer Engineering, Univ. of Texas at Austin, May 1999.
- [5] MATLAB and Simulink for Technical Computing, URL: <http://www.mathworks.com>, July 2010.
- [6] Tarun Kumar, Karun Verma, "A Theory Based on Conversion of RGB image to Gray image", International Journal of

Computer Applications (0975 – 8887) Volume 7– No.2, September 2010.

- [7] X.L. Zhang —Digital image edge detection with MATLAB I. Journal of Jilin institute of chemical technology. Vol.27, no.2, pp.59-61, 2010.
- [8] Ravi Kumar A.V, Nataraj K.R, "Result Analysis of LabVIEW and MATLAB in Application of Image Edge Detection" International Journal of Computer Applications (0975 – 888) Volume 48– No.9, June 2012
- [9] Edge DetectionI, chapter of —NI Vision Concepts Help, online Resource.