

A Novel Edge Detection Technique for Image Classification and Analysis

Mr.Srinivasa Rao Elisala¹, Y.Smruthi², B.Lavanya³, E.Nageswara Rao⁴

Abstract: The main aim of this project is to propose a new method for image segmentation. Image Segmentation is concerned with splitting an image up into segments (also called regions or areas) that each holds some property distinct from their neighbor. Simply, another word for the Object Detection is "Segmentation". Segmentation is divided into two types they are Supervised Segmentation and Unsupervised Segmentation. Segmentation consists of three types of methods which are divided on the basis of threshold, edge and region. Thresholding is a commonly used enhancement whose goal is to segment an image into object and background. Edge-based segmentations rely on edges found in an image by edge detecting operators. Region based segmentations basic idea is to divide an image into zones of maximum homogeneity, where homogeneity is an important property of regions. Edge detection has been a field of fundamental importance in digital image processing research. Edge can be defined as a pixels located at points where abrupt changes in gray level take place in this paper one novel approach for edge detection in gray scale images, which is based on diagonal pixels in 2*2 region of the image, is proposed. This method first uses a threshold value to segment the image and binary image. And then the proposed edge detector. In order to validate the results, seven different kinds of test images are considered to examine the versatility of the proposed edge detector. It has been observed that the proposed edge detector works effectively for different gray scale digital images. The results of this study are quite promising. In this project we proposed a new algorithm for edge Detection. The main advantage of this algorithm is with running mask on the original image we can detect the edges in the images by using the proposed scheme for edge detection.

Keywords: Edge detection, segmentation, thresholding.

I. Introduction

Types of Segmentation

1. Thresholding:

Thresholding has a long but checkered history in digital image processing [9]. Thresholding is a commonly used enhancement [12] whose goal is to segment an image into *object* and *background*. A threshold value is computed above (or below) which pixels are considered "object" and below (or above) which "background", and eliminates unimportant shading variation. One obvious way to extract [6] the objects from the background is to select a threshold **T** that separates these modes. Then, any point (x, y) for which $f(x, y) > T$ is called an object point; otherwise, the point is called a background point. Thresholding [3] can also be done using neighborhood operations. Thresholding essentially involves turning a color or grayscale image into a 1-bit binary image. This is done by allocating every pixel in the image either black or white, depending on their value. The pivotal value that is used to decide whether any given pixel is to be black or white is the threshold. If the threshold value is only dependent on the gray values then it is called as "**Global Thresholding**". If the threshold value is dependent on the gray values and on some local property then it is called as "**Local Thresholding**". If the threshold value is dependent on the gray values, some local property and some spatial coordinates then it is called as "**Adaptive Thresholding**". Different ways to choose the threshold value when we convert a gray image to binary image are;

1. Randomly selecting the gray values of the image
2. Average of all the gray values of the image:
3. Median of given gray values:
4. (Minimum value + Maximum value)/2

Algorithm for Selecting the Threshold value randomly:

Step 1: Read image.

Step 2: Convert image into two dimensional matrixes (eg: 64X64).

Step 3: Select the threshold value randomly.

Step 4: If grey level value is greater than the threshold value makes it bright, else make it dark.

Step 5: Display threshold image.

Algorithm for Selecting the Threshold value using Average of all gray values:

- Step 1: Read image.
- Step 2: Convert image into two dimensional matrixes (eg: 64X64).
- Step 3: Select the threshold value by calculating average of all the gray values.
- Step 4: If grey level value is greater than the threshold value makes it bright, else make it dark.
- Step 5: Display threshold image.

Algorithm for Selecting the Threshold value using Median of all gray values:

- Step 1: Read image.
- Step 2: Convert image into two dimensional matrixes (eg: 64X64).
- Step 3: Select the threshold value by calculating median of all the gray values.
- Step 4: If grey level value is greater than the threshold value make it bright, else make it dark.
- Step 5: Display threshold image.

Algorithm for Selecting the Threshold value using Average of minimum and maximum gray values:

- Step 1: Read image.
- Step 2: Convert image into two dimensional matrixes (eg: 64X64).
- Step 3: Select the threshold value by calculating the average of Minimum and Maximum gray values.
- Step 4: If grey level value is greater than the threshold value makes it bright, make it dark.
- Step 5: Display threshold image.

Limitations:

- 1. As hardware cost dropped and sophisticated new algorithms were developed, thresholding became less important.
- 2. Thresholding destroys useful shading information and applies essentially infinite gain to noise at the threshold value, resulting in a significant loss of robustness and accuracy.

II. Edge Based Segmentation:

The edges of an image hold much information in that image. The edges tell where objects are, their shape and size, and something about their texture. An edge is where the intensity of an image [10] moves from a low value to a high value or vice versa. Edges in images are areas with strong intensity contrasts – a jump in intensity from one pixel to the next. Edge detecting [4] is an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image. Edge detectors are a collection of very important local image pre-processing methods [11] used to locate (sharp) changes in the intensity function. Edges are pixels where the brightness function changes abruptly. Edge-based segmentation [5] represents a large group of methods based on information about edges in the image; it is one of the earliest segmentation approaches and still remains very important and also relies on edges found in an image by edge detecting operators these edges mark image locations [8] of discontinuities in gray level, color, texture, etc. The classification [1] of edge detectors is based on the behavioral study of the edges with respect to the following operators:

- (i).First Order derivative edge detection
- (ii).Second Order derivative edge detection

(i).First Order edge detection Technique:

The first derivative assumes a local maximum at an edge. For a gradient image $f(x, y)$, at location (x, y) , where x and y are the row and column coordinates respectively, we typically consider the two directional derivatives. The two functions that can be expressed in terms of the directional derivatives [3] are the gradient magnitude and the gradient orientation. The gradient magnitude is defined by,

$$|\nabla f| = \left[\begin{matrix} G_x \\ G_y \end{matrix} \right] = \left[\begin{matrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{matrix} \right] = [G_x^2 + G_y^2]^{\frac{1}{2}}$$

The gradient orientation is also an important quantity. The gradient orientation is given by,

$$\angle \nabla f(x, y) = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

The angle is measured with respect to the x- axis. The other method of calculating the gradient is given by estimating the finite difference.

$$\frac{\partial f}{\partial x} = \lim_{h \rightarrow 0} \frac{f(x+h, y) - f(x, y)}{h}$$

$$\frac{\partial f}{\partial y} = \lim_{h \rightarrow 0} \frac{f(x, y+h) - f(x, y)}{h}$$

We can approximate this finite difference as;

$$\frac{\partial f}{\partial x} = \frac{f(x+h, y) - f(x, y)}{h_x} = f(x+1, y) - f(x, y), (h_x = 1)$$

$$\frac{\partial f}{\partial y} = \frac{f(x, y+h) - f(x, y)}{h_y} = f(x, y+1) - f(x, y), (h_y = 1)$$

(ii).Second Order edge detection technique:

Second Order derivative [7] edge detection techniques employ some form of spatial second order differentiation [5] to accentuate edges. An edge is marked if a significant spatial change occurs in the second derivative.

Laplacian and Gaussian Edge detection operator:

The Laplacian operator is a very popular operator approximating the second derivative which gives the gradient magnitude only. The principle used in the Laplacian of Gaussian method is, the second derivative of a signal is zero when the magnitude of the derivative is maximum. The Laplacian is approximated in digital images by a convolution sum. The Laplacian of a 2-D function $f(x, y)$ is defined as;

$$(\nabla^2 f)(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

3x3 masks for 4-neighborhoods and 8-neighborhood for this operator are as follows;

$$G_x = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad G_y = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

The two partial derivative approximations for the Laplacian for a 3x3 region are given as,

$$\nabla^2 f = 4(a_8) - (a_1 + a_3 + a_5 + a_7)$$

$$\nabla^2 f = 8(a_8) - (a_0 + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 + a_7)$$

The 5x5 Laplacian is a convoluted mask to approximate the second derivative. Laplacian uses a 5x5 mask for the 2nd derivative in both the x and y directions.

-1	-1	-1	-1	-1
-1	-1	-1	-1	-1
-1	-1	24	-1	-1
-1	-1	-1	-1	-1
-1	-1	-1	-1	-1

ALGORITHM FOR LAPLACIAN AND GAUSSIAN OPERATOR:

- Step 1: Read image.
- Step 2: Convert image into two dimensional matrix (eg: 64X64).
- Step 3: Calculate Threshold value of the image using average method.
- Step 4: If grey level value is greater than or equal to the threshold value make it bright, else make it dark.
- Step 5: Display Threshold image.
- Step 6: Perform Gaussian Operation on Original image to reduce the noise.
- Step 7: Take 3x3 Laplacian Operator and place it on the Original Image which is masked by the Gaussian operator.
- Step 7: Find the Gradient Magnitude and replace the center pixel by the magnitude.
- Step 8: Convolve the mask on the entire image.
- Step 9: Display the Convolved image.

Limitations:

- 1. It responds doubly to some edges in the image.

Limitations for Edge Based Segmentation:

Edge-based methods center around contour detection: their weakness in connecting together broken contour lines make them, too, prone to failure in the presence of blurring. The main disadvantage of these edge detectors is their dependence on the size of objects and sensitivity to noise. Further, since conventional boundary[2] finding relies on changes in the grey level, rather than their actual values, it is less sensitive to changes in the grey scale distributions over images as against region based segmentation.

III. Proposed Method

Digital image processing is a general term for the wide range of techniques that exist for interpreting or modifying digital images in different ways. Principle application of digital image processing is machine perception of visual information, as used in robotics. An image may be defined as a 2D function $f(x, y)$, where x and y are spatial coordinates and amplitude of f at any pair of coordinates (x, y) is called the intensity or gray level of the image at that point. When all values of x, y and f are finite and discrete then such image is called as Digital Image. A digital image is defined by a finite values function over a discrete domain z^2 suppose the digital image domain D , which is proper subset of z^2 , is a matrix of size $M \times N$

i.e., $D = \{(x, y): x=1, 2, 3 \dots M-1, M; y=1, 2, 3 \dots N-1, N\}$

Obtained by sampling with unit step along x and y directions. So a digital image $f(x, y)$ can be represented by an $M \times N$ matrix whose elements are integer numbers ranging from 0 to $L-1$:

$f(1,1)$	$f(1,2)$	$f(1,3)$	\dots	$f(1,N-1)$	$f(1,N)$
$f(2,1)$	$f(2,2)$	$f(2,3)$	\dots	$f(2,N-1)$	$f(2,N)$

\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots

$f(M,1)$	$f(M,2)$	$f(M,3)$	\dots	$f(M,N-1)$	$f(M,N)$
----------	----------	----------	---------	------------	----------

Each element of matrix is called pixel. Thus, the pixel may indicate only a location in z^2 (i.e. the coordinate), or it may represent the gray level along with position. For gray scale image $L=256$.

An image may be considered to contain sub images sometimes referred to as regions of interest (ROIs) or simply regions. Middle level of digital image processing involves tasks such as segmentation (partitioning an image into regions or objects), description of those objects to reduce them to a form suitable for computer processing [10] and classification (recognition) of individual objects. In this level of digital image processing, inputs are images, but outputs are attributes extracted from those images such as edges. Edges can be defined as locations where the values of adjacent points differ sufficiently.

Edges in the paradigm of image processing and computer vision provide valuable information towards human image understanding. Thus, edges play important role in human picture recognition system. Naturally, edge detection has become a serious challenge to the image processing scientists, and since the last two decades, in particular, numerous publications have been detailing methodologies for edge detection. Edges are the representations of the discontinuities of image intensity function. Edge detection algorithm is a process of detection of these discontinuities in an image. Edge detecting an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image. There are many ways to perform edge detection like the Gradient method and the Laplacian method. The Gradient method detects the edges by looking for the maximum and minimum in the first derivation of the image. The Laplacian method searches for zero crossings in the second derivation of the image to find edges.

SELECTION OF THRESHOLD VALUE

Image Thresholding is an image processing technique in which the points of an image at or near a particular value are highlighted. When a threshold value is applied onto a gray scale image, this highlighting can be done by converting all pixels below threshold value to black, and all pixels above that value to white. Thus a threshold edge is produced between the blacks and white regions. Because there are two possible output values, thresholding creates a binary image. If T is a threshold value, then any pixel (x, y) for which $f(x, y) > T$ is called an object point; otherwise the pixel is called a background pixel.

In general, the threshold can be chosen as the relation, $T = T[x, y, p(x, y), f(x, y)]$ where $f(x, y)$ is the gray level of the pixel (x, y) and $p(x, y)$ denotes some local property of this pixel - for example, the average gray level of a neighborhood centered on (x, y) . A threshold image $h(x, y)$ is defined as $h(x, y) = 1$ if $f(x, y) > T$; otherwise $h(x, y) = 0$. Thus, pixels labeled 1 correspond to objects, whereas pixels labeled 0 correspond to the background. When T depends only on $f(x, y)$ (i.e., only on gray level values), the threshold is called global. If T depends on $f(x, y)$ and $p(x, y)$, the threshold is called local. If T depends on the pixel position (x, y) as well as $f(x, y)$ at that pixel position, then it is called dynamic or adaptive threshold. In the proposed scheme to detect edges, global threshold value is used.

Procedure to select suitable threshold value

Step1. Select an initial estimate for T .

Step2. Segment the image using T . This will produce two groups of pixels: R_1 consisting of all pixels with gray level values $> T$ and R_2 consisting of pixels with gray level values $\leq T$.

Step3. Compute the average gray level value and μ_2 for the pixels in region R_1 and R_2 .

Step4. compute a new threshold value Set $T_{new} = (\mu_1 + \mu_2) / 2$, and Set $T_{old} = 0$

Step5: while $(T_{new} \neq T_{old})$ do $\mu_1 =$ Mean gray level of pixels for which $f(x, y) \geq T_{new}$ $\mu_2 =$ Mean gray level of pixels for which $f(x, y) < T_{new}$ Set $T_{old} = T_{new}$

Set $T_{new} = (\mu_1 + \mu_2) / 2$ End while

Step6: suitable threshold value Set $T = T_{new}$

Step7: Stop in digital image processing, an image defined in the real world is considered to be a function of to real variables, for example,

$f(x, y)$ with f as the amplitude of the image at the real coordinate position (x, y) .

A spatial filter mask may be defined as a (template) matrix w of size $m \times n$. Assume that $m = 2a$ and $n = 2b$, where a, b are nonzero positive integers.

Smallest meaningful size of the mask is 2×2 . Such mask coefficients, showing coordinate arrangement as:

Image region under the above mask is shown below

$W(0,0)$	$W(0,1)$
$W(1,0)$	$W(1,1)$

$f(x,y)$	$f(x,y+1)$
$f(x+1,y)$	$f(x+1,y+1)$

Basic idea behind edge detection:

a) Classification of all pixels that satisfy the criterion of homogeneousness.

b) Detection of all pixels on the borders between different homogeneous areas. In the proposed scheme, first create a binary image by choosing a suitable threshold value. Filter mask is applied on the binary image to detect edge pixels. Set Filter mask coefficients $w(0,0), w(1,1)$ equal to 1 and $w(0,1), w(1,0)$ equal to \times , as shown below:

1	\times
\times	1

Move the filter mask on the whole image and find the gray level value of each pixel of the image under the \times of the filter mask in each masking step. If homogeneousness does not exist between neighbors (except diagonal pixel) of the pixel that under the \times of the mask, the pixel is an edge pixel, otherwise not. By taking decision

about the two pixels in the mask, major problem of processing border pixels in the image is solved. Only two pixels (first and last) in the whole image are left. These two pixels have to be processed separately.

Proposed Algorithm

Step1. Create a binary image by choosing a suitable threshold value.

If $(f(x, y) > \text{threshold value})$, then
 Set $f(x, y) = 1$, otherwise
 Set $f(x, y) = 0$ End if

Step2. Find edge pixel in binary image:

2.1. Create a mask, w , with dimensions 2×2 . Set $w(0, 0) = w(1, 1) = 1$
 Set $w(1, 0) = w(0, 1) = \times$

2.2 .Create an $M \times N$ output image, g :

For all pixel coordinates, x and y , do

Set $g(x, y) = f(x, y)$ End for

2.3. Checking for edge pixels

2.3.1. Process whole image except first and last pixels

For $y=1$ to $N-1$ do for $x=1$ to $M-1$ do If (gray level values of $f(x, y+1)$, $f(x, y)$ And $f(x+1, y+1)$ are equal) Then, $f(x+1, y)$ is not an edge pixel, And Set $g(x, y+1) = 0$ Otherwise $(x+1, y)$ is an edge pixel And Set $g(x, y+1) = 1$ End if End for

Process first pixel of the image

If (gray level values of $f(1, 1)$, $f(1, 2)$ and $f(2, 1)$ are equal),

Then $f(1, 1)$ is not an edge pixel, and

Set $g(1,1)=0$ otherwise $f(1,1)$ is an edge pixel And Set $g(1,1)=1$ End if process last pixel of the image gray level values of $f(M,N)$, $f(M,N-1)$ and

$f(M-1, N)$ are equal) then $f(M,N)$ is not an edge pixel, And Set $g(M,N)=1$

End if

Step3.stop

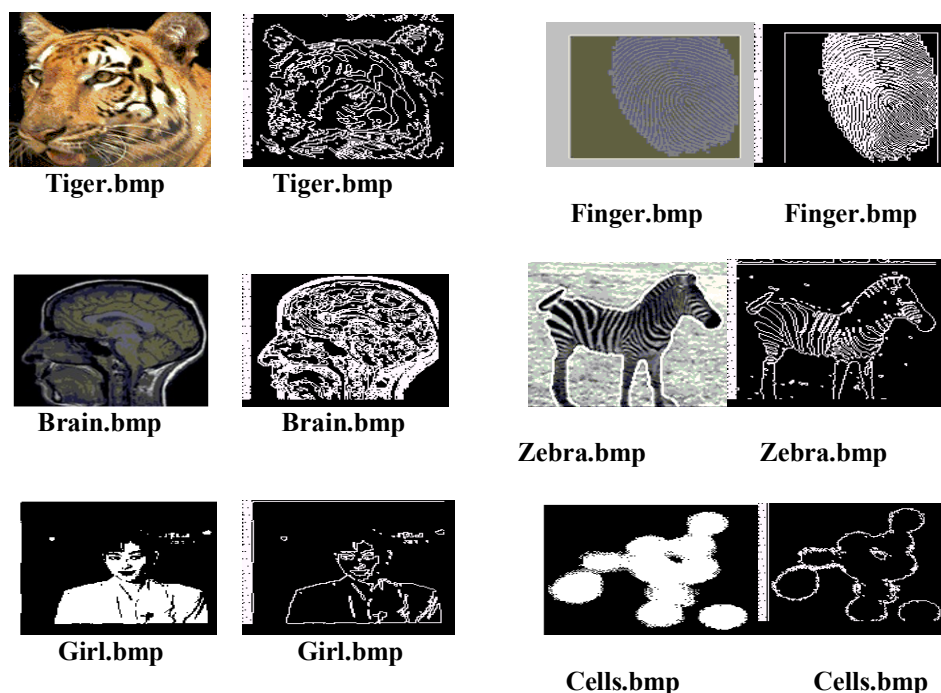
IV.Result Discussions

First, in order to evaluate the performance of the proposed scheme for edge detection, a standard test image of brain was taken. Global threshold value was calculated for that image using above procedure for calculating suitable threshold value on its edge was detected using proposed procedure. Threshold value for that image is 0.3472 when image in normalized form (i.e., all gray level values lie between 0 and 1). The result of edge detection is shown in figure1. It has been observed that the proposed method for edge detection works well. In order to validate the results about the performance of proposed scheme for edge detection, seven different test images are considered. Global threshold values calculated by the threshold evaluation procedure for different test images are given in table 1. The results of edge detections for these test images are shown in figure2. From the results; it has again been observed that the performance of the proposed edge detection scheme is found to be satisfactory for all the test images.

S.NO	Image	Threshold value
1	Tiger	0.075830
2	Finger print	0.083817
3	Skull	0.040524
4	Zebra	0.121945
5	Girl	0.47811
6	Cells	0.61961

Table1: Threshold values for different standard images.

Original images & Output images



V. Conclusions

Computer vision and digital image processing are rapidly growing fields that are important in numerous aspects in many fields. Until a few years ago, segmentation techniques were proposed mainly for gray-level images since for a long time these have been the only kind of visual information that acquisition devices were able to take and computer resources to handle. The main goal of the image segmentation is to divide an image into parts that have a strong correlation with objects of the real world depicted in the image. Thresholding represents the simplest image segmentation process, and it is computationally inexpensive and fast. Edge based segmentation relies on edges found in an image by edge detecting operators. Gradient operators determine edge location in which the image function under goes rapid changes, they have a similar effect to suppress low frequencies in the Fourier Transform Domain. Most Gradient operators can be expressed using Convolution masks like Robert, Prewitt, Sobel, Robinson, and Krisch. Zero crossing of the second derivative are most robust than small-size gradient detectors and calculated Laplacian of Gaussian. In our proposed method, an attempt is made to develop as a new technique for edge detection. Experimental results have demonstrated that the proposed scheme for edge detection works satisfactorily for different gray level digital images. The theoretical principles and systematic development of the algorithm for the proposed versatile edge detector is described in detail. The technique has potential future in the field of digital image processing. The work is under further progress to examine the performance of the proposed edge detector for different gray level images affected with different kinds of noise.

References

- [1]. C.Naga Raju , O.Rama Devi, Sharada Mani "An Improved Ant Colony Optimization Technique by using Fuzzy Inference Rules for Image Classification and Analysis" International Journal of Advanced Engineering & Applications. Jan. 2010.
- [2]. C.NagaRaju and Dr.L.S.S.Reddy "Novel Watershed Segmentation Method for Stumpy Boundary Detection for Image Classification" World academy of science, engineering and Technology International Journal of Information Technology and Knowledge Management Volume 2, No. 1, pp. 177-180 January June 2009.
- [3]. Dr.LSS Reddy, A.Rama Swami Reddy, C.NagaRaju and B.Sreenivasa Rao" generation of random fields based on matrix-geometric solution method for image analysis and classification" International Journal of Advanced Engineering & Applications, Italy. Jan. 2010.
- [4]. Sharada Mani, G. SyamPrasad, O.Rama Devi, C.NagaRaju "A Novel Edge Detection Technique for Color Images" International Journal of Advanced Engineering & Applications Jan. 2010.

- [5]. Shahana Bano, M. Surendra Prasad Babu, C.Naga Raju” A New Proposed Method for Image Segmentation Based on Gray Scale Morphological Transformations” International Journal of Computer Science and Technology IJCST Vol. 1, Issue 2,December 2010.
- [6]. C.Nagaraju, B.Srinu, E.Srinivasa Rao “An efficient Facial Features extraction Technique for Face Recognition system Using Local Binary Patterns” International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-2, Issue-6, May 2013.
- [7]. R.M. Haralick ,Digital step Ridges from zero crossings of second directional derivatives, IEEE transactions on Pattern Analysis and Machine Intelligence, January (1984).
- [8]. A.M.Baumberg and D C Hogg. Learning spatiotemporal models from examples. Image and Vision Computing, 14(8):525-532, 1996.
- [9]. Rafael c. Gonzales and Richard E. Woods. ‘Digital image processing ‘. Pearson Education, Delhi, 2004.
- [10]. Chanda and D.Dutta Majumder. ‘Digital image processing and Analysis ‘. Prentice-Hall of India, New Delhi, 2002.
- [11]. Anil k.jain Fundamentals of digital image processing and Analysis Prentice hall of India New Delhi, 2003.
- [12]. Dwayne Phillips image processing: Analyzing and enhancing digital images BPB, New Delhi, 2002.

About The Authors



Mr. E. Srinivasa Rao received his B.Tech degree in Information Technology from Laki Reddy Bali Reddy College of Engineering Mylavaram, M.Tech in Software Engineering from Gayatri Vidya Parishad College of Engineering Visakhapatnam. He has published 3 International journals and he has attended one workshop.



Mrs.Y.Smruthi received her B.Tech degree in electronics and communication engineering from Bhoji reddy engineering college for women Hyderabad, M.Tech in electronics and communications from Geetanjali College of engineering and technology, Hyderabad. . Currently she is working as Asst.professor in Vignan Bharathi Institute of Technology, Hyderabad.



Mrs.B.Lavanya received her B.Tech degree in electronics and communication engineering from MRITS, Udayagiri M.Tech in electronics and communications from Geetanjali College of engineering and technology, Hyderabad. Currently she is working as Asst.professor in Vignan Bharathi Institute of Technology, Hyderabad.



Mr. E. Nageswara Rao received his B.Sc degree in Electronics Andhra Loyola College, Vijayawada, M.Sc (Tech) in Geophysics from Andhra University, Visakhapatnam. And he is Pursuing Ph.D in Geophysics from Andhra University, Visakhapatnam. Currently, he is working as a CSIR-Research Intern in NGRI, Hyderabad.