

Detection of Leg Fracture in X-Ray Images using Hough Transform

A Project report submitted in partial fulfillment of the requirements for

the award of the degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING

By

B. Deva Harshitha (316126512004)

D.V. Guru Saran (316126512013)

Y. Ravi Teja (316126512060)

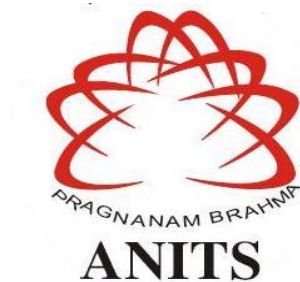
E. Manohar (316126512017)

Under the Esteemed Guidance of

Mr. A. SIVA KUMAR, MTech (PhD)

Assistant professor

Department of ECE, ANITS



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(UGC AUTONOMOUS)

(Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade)

Sangivalasa, Bheemili Mandal, Visakhapatnam dist. (A.P)

2019-2020

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES
(UGC AUTONOMOUS)
(Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade)
Sangivalasa, Bheemili Mandal, Visakhapatnam dist. (A.P)



CERTIFICATE

*This is to certify that the industrial training report entitled “**Detection of Leg Fracture in X-Ray Images using Hough Transform**” submitted by **B. Deva Harshitha-316126512004, D.V.Guru Saran-316126512013, Y. Ravi Teja-316126512060, E. Manohar-316126512017** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology in Electronics & Communication Engineering** of Andhra University; Visakhapatnam is a record of bonafide work carried out under my supervision.*

UNDER THE GUIDANCE OF

HEAD OF THE DEPARTMENT

Mr. A. SIVA KUMAR MTech., (Ph.D.)
Assistant professor
Department of ECE
ANITS

Dr.V. Rajya Lakshmi
M.E., Ph.D., MHRM, MIEEE, MIE, MIETE
Department of ECE
ANITS

ACKNOWLEDGEMENT

We are grateful to **Dr. V. Rajya Lakshmi**, Head of the Department, Electronics and Communication Engineering, for providing us permission and with the required facilities for the completion of the industrial training work.

We are very much thankful to the **Principal and Management, ANITS, Sangivalasa**, for their encouragement and cooperation to carry out this work.

We would like to express our deep gratitude to **Mr. A. SIVA KUMAR, Assistant professor Department of ECE, ANITS** for his guidance. We express our thanks to all teaching faculty of Department of ECE, for their encouragement helped using accomplishment of our industrial training.

We would like to thank our parents, friends, and classmates for their encouragement throughout our industrial training period. At last, but not the least, we thank everyone for supporting us directly or indirectly in completing this industrial training successfully.

Project Student:

B. Deva Harshitha (316126512004)

D.V. Guru Saran (316126512013)

Y. Ravi Teja (316126512060)

E. Manohar (316126512017)

ABSTRACT

The bone fracture is a common problem in human beings occurs due to high pressure is applied on bone or simple accident and also due to osteoporosis and bone cancer. The image processing techniques are very useful for many applications such as biology, security, satellite imagery, personal photo, medicine, etc. The procedures of image processing such as image enhancement, image segmentation and feature extraction are used for fracture detection system. In this project we use Canny edge detection method for segmentation. Canny method produces perfect information from the bone image. The main aim of this project is to detect human lower leg bone fracture from X-Ray images. The proposed system has three steps, namely, preprocessing, segmentation, and fracture detection. In feature extraction step, this paper uses Hough transform technique for line detection in the image. The results from various simulation show that the proposed system is very accurate and efficient.

CONTENTS

ABSTRACT	6
LIST OF FIGURES	7
LIST OF ABBREVIATIONS	8
CHAPTER 1 Introduction	9
1.1 Digital Image Processing System	9
1.1.1 Image processing system	9
1.1.2 Image processing fundamentals	11
1.1.2.1 Fundamental steps in Image processing	11
1.1.2.2 Image Types	13
1.1.2.3 Image processing Goals	14
1.1.2.4 Applications of Image processing	17
1.2 Objective	18
1.3 Existing System	18
1.3.1 Disadvantages of Existing System	18
1.4 LITERATURE SURVEY	19
1.5 Proposed System	23
1.5.1 Proposed System Block diagram	23
1.5.2 Proposed System Advantages	24
CHAPTER 2 Project Description	25
2.1 Introduction	25
2.2 General	26
2.2.1 Preprocessing	26
2.3 Edge detection Techniques	27
2.3.1 Robert edge detection	28
2.3.2 Sobel edge detection	28
2.3.3 Prewitt edge detection	29
2.3.4 Canny edge detection	29

2.3.5 LoG edge detection	30
2.3.6 Result	30
2.4 Hough transform	32
2.4.1 Introduction	32
2.4.2 Hough-transform – the input	33
2.4.3 Input to Hough-thresholded edge image	34
2.4.4 Hough transform basic idea	36
2.4.4.1 Hough transform- algorithm	37
2.4.4.2 Hough transform-polar representation of lines	38
2.4.4.3 Hough transform-algorithm using polar representation of lines	40
2.4.5 Advantages and Disadvantages	43
2.5 Methodologies	43
2.5.1 Module Names	43
CHAPTER 3 Software Specifications	45
3.1 Introduction	45
3.2 Features in MATLAB	46
3.2.1 Interfacing with other Languages	47
3.2.2 Analyzing and Accessing data	49
3.2.3 Performing Numeric Computation	51
CHAPTER 4 Results	52
4.1 Implementation	52
4.2 Snapshots	52
4.2.1 Original X-ray images	53
4.2.2 Filtered Image	54
4.2.3 Edge detection image	55
4.2.4 Hough-Lines image	56
4.2.5 Hough-Peaks image	57
4.2.6 Hough-Thresholded image	58
4.2.7 Final Result	59
CHAPTER 5 Applications	60
CHAPTER 6 Conclusion and REFERENCES	61

LIST OF FIGURES

Fig 1.1	Block Diagram for Image Processing System	9
Fig 1.2	Block Diagram of Fundamental Sequence involved in image processing system	14
Fig 1.3	Detailed Block Diagram of Proposed System	24
Fig 2.1	Structure of Lower Leg Bone	25
Fig 2.2	Results of Image preprocessing	27
Fig 2.3	Various edge detection technique images	31
Fig 2.4	Linear Structured image before and after performing Edge Detection	32
Fig 2.5	Edge Detected Image and Thresholded image	35
Fig 2.6	Plot of Different Lines passing through a Single point	36
Fig 2.7	Plot Showing the transformation of a single point	37
Fig 2.8	Hough Accumulator Cells	38
Fig 2.9	Polar Representation of lines	39
Fig 2.10	Original and Sobel Edge Detected Image	41
Fig 2.11	Original and Thresholded Image	42
Fig 2.12	Original and Hough Transformed Image	42

LIST OF ABBREVTIONS

CD-ROM	Compact Disc Read-Only Memory	10
CRTs	Cathode ray tubes	11
LCDs	Liquid crystal displays	11
CCD	Charged Coupled Device	12
GIS	geographic information system	13
CT	Computed Tomography	13
NMR	Nuclear magnetic resonance	13
MRI	Magnetic Resonance Imaging	17
CAT	Computerized Axial Tomography	17
DICOM	Digital Imaging and Communications in Medicine	18
PACS	Picture Archives and Communication System	18
MATLAB	Matrix Laboratory	25
JMI	Java-to-Mat lab Interface	47
GUIDE	Graphical User Interface Development Environment	49

CHAPTER 1

INTRODUCTION

Digital image processing is the use of computer algorithms to perform image processing on digital images. The 2D continuous image *is* divided into N rows and M columns. The intersection of a row and a column is called a pixel. The image can also be a function other variable including depth, color, and time. An image given in the form of a transparency, slide, photograph or an X-ray is first digitized and stored as a matrix of binary digits in computer memory. This digitized image can then be processed and/or displayed on a high-resolution television monitor. For display, the image is stored in a rapid-access buffer memory, which refreshes the monitor at a rate of 25 frames per second to produce a visually continuous display.

1.1.1 THE IMAGE PROCESSING SYSTEM

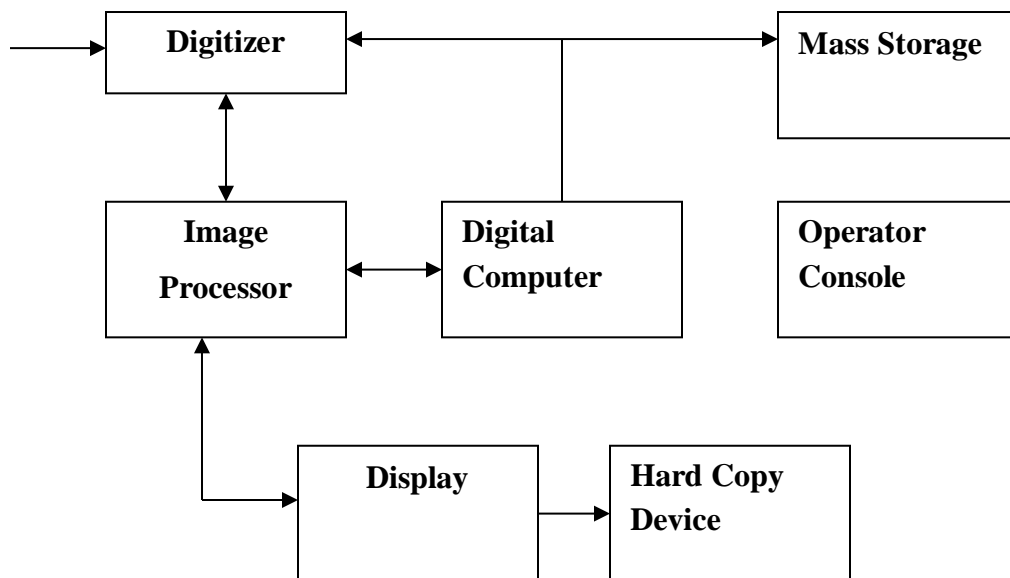


FIG 1.1 BLOCK DIAGRAM FOR IMAGE PROCESSING SYSTEM

➤ **DIGITIZER**

Digitizing or digitization is the representation of an object, image, sound, document or a signal (usually an analog signal) by a discrete set of its points or samples. Digital information exists as one of two digits, either 0 or 1. These are known as bits.

An image is digitized to convert it to a form which can be stored in a computer's memory or on some form of storage media such as a hard disk or CD-ROM. This digitization procedure can be done by a scanner, or by a video camera connected to a frame grabber board in a computer. Once the image has been digitized, it can be operated upon by various image processing operations.

- Microdensitometer
- Flying spot scanner
- Image dissector
- Videocon camera
- Photosensitive solid- state arrays.

➤ **DIGITAL COMPUTER**

A computer is an electronic device that accepts raw data, processes it according to a set of instructions and required to produce the desired result. Mathematical processing of the digitized image such as convolution, averaging, addition, subtraction, etc. are done by the computer.

➤ **MASS STORAGE**

Mass storage devices used in desktop and most server computers typically have their data organized in a file system. The secondary storage devices normally used are floppy disks, CD ROMs etc.

➤ **OPERATOR CONSOLE**

The operator console consists of equipment and arrangements for verification of intermediate results and for alterations in the software as and when require. The operator is also capable of checking for any resulting errors and for the entry of requisite data.

➤ **DISPLAY**

Popular display devices produce spots (display elements) for each pixel:

- ❖ Cathode ray tubes (CRTs).
- ❖ Liquid crystal displays (LCDs).
- ❖ Printers.

Spots may be binary (e.g., monochrome LCD), achromatic (e.g., so-called black-and-white, actually grayscale for intensity), pseudo color or false colors (e.g., for intensity or hyper spectral data), or true color (color data displayed as such).

1.1.2 IMAGE PROCESSING FUNDAMENTAL

Digital image processing refers processing of the image in digital form. Modern cameras may directly take the image in digital form but generally images are originated in optical form. They are captured by video cameras and digitalized. The digitalization process includes sampling, quantization. Then these images are processed by the five fundamental processes, at least any one of them, not necessarily all of them.

1.1.2.1 FUNDAMENTAL STEPS IN IMAGE PROCESSING

1. Image acquisition
2. Image preprocessing
3. Image segmentation
4. Image representation
5. Image description
6. Image recognition
7. Image interpretation

➤ **IMAGE ACQUISITION**

First, we need to produce a digital image from a paper envelope. This can be done using either a CCD camera, or a scanner

➤ **IMAGE PREPROCESSING**

This is the step taken before the major image processing task. The problem here is to perform some basic tasks in order to render the resulting image more suitable for the job to follow. In this case it may involve enhancing the contrast, removing noise, or identifying regions likely to contain the postcode.

➤ **IMAGE SEGMENTATION**

Segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

➤ **IMAGE REPRESENTATION**

Image process is the process of convert the input data to a form suitable for computer processing

➤ **IMAGE DESCRIPTION**

Image description is the process of extract features that result in some quantitative information of interest or features that are basic for differentiating one class of objects from another.

➤ **IMAGE RECOGNITION**

Image recognition is the process of assign a label to an object based on the information provided by its descriptors.

➤ **IMAGE INTERPRETATION**

Image interpretation is the process of assign meaning to an ensemble of recognized objects.

1.1.2.2 IMAGE TYPES

There are several ways of encoding the information in an image.

1. Binary image
2. Grayscale image
3. Indexed image
4. True color or RGB image

➤ **BINARY IMAGE**

Each pixel is just black or white. Since there are only two possible values for each pixel (0, 1), we only need one bit per pixel.

➤ **GRAYSCALE IMAGE**

Each pixel is a shade of gray, normally from 0 (black) to 255 (white). This range means that each pixel can be represented by eight bits, or exactly one byte. Other grayscale ranges are used, but generally they are a power of 2.

➤ **INDEXED IMAGE**

An indexed image consists of an array and a color map matrix. The pixel values in the array are direct indices into a color map. By convention, this documentation uses the variable name *X* to refer to the array and *map* to refer to the color map.

➤ **TRUE COLOR OR RGB IMAGE**

Each pixel has a particular color; that color is described by the amount of red, green and blue in it. If each of these components has a range 0–255, this gives a totally of 256³ different possible colors. Such an image is a “stack” of three matrices; representing the red, green and blue values for each pixel. This means that for every pixel there correspond 3 values.

1.1.2.3 IMAGE PROCESSING GOALS

In virtually all image processing applications, however, the goal is to extract information from the image data. Obtaining the information desired may require filtering, transforming, coloring, interactive analysis, or any number of other methods.

To be somewhat more specific, one can generalize most image processing tasks to be characterized by one of the following categories:

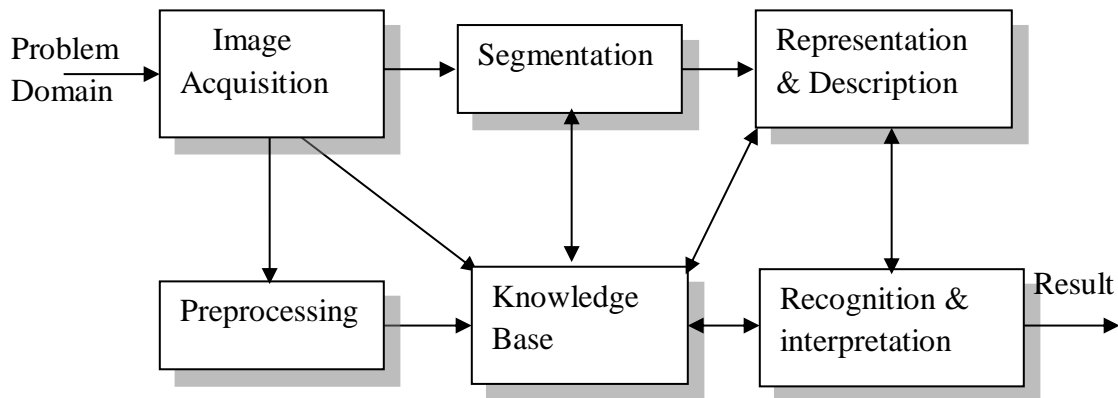


FIG 1.2 BLOCK DIAGRAM OF FUNDAMENTAL SEQUENCE INVOLVED IN AN IMAGE PROCESSING SYSTEM

1. Image enhancement
2. Image restoration
3. Image analysis
4. Feature extraction
5. Image registration
6. Image compression
7. Image synthesis

➤ **IMAGE ENHANCEMENT**

This simply means improvement of the image being viewed to the (machine or human) interpreter's visual system. Image enhancement types of operations include contrast adjustment, noise suppression filtering, application of pseudo color, edge enhancement, and many others.

➤ **IMAGE RESTORATION**

The purpose of image restoration is to "compensate for" or "undo" defects which degrade an image. Degradation comes in many forms such as motion blur, noise, and camera misfocus. In cases like motion blur, it is possible to come up with a very good estimate of the actual blurring function and "undo" the blur to restore the original image. In cases where the image is corrupted by noise, the best we may hope to do is to compensate for the degradation it caused.

➤ **IMAGE ANALYSIS**

Image analysis is the extraction of meaningful information from images. Image analysis operations produce numerical or graphical information based on characteristics of the original image. They break into objects and then classify them. They depend on the image statistics. Common operations are extraction and description of scene and image features, automated measurements, and object classification. Image analyze are mainly used in machine vision applications.

➤ **FEATURE EXTRACTION**

Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. When performing analysis of complex data one of the major problem's stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power or a classification algorithm which over fits the training sample and generalizes poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

➤ **IMAGE REGISTRATION**

Image registration is the process of overlaying two or more images of the same scene taken at different times, from different viewpoints, and/or by different sensors. It geometrically aligns two images the reference and sensed images. The present differences between images are introduced due to different imaging conditions. Image registration is a crucial step in all image analysis tasks in which the final information is gained from the combination of various data sources like in image fusion, change detection, and multichannel image restoration.

Typically, registration is required in remote sensing (multispectral classification, environmental monitoring, change detection, image mosaicking, weather forecasting, creating super-resolution images, integrating information into geographic information systems (GIS)), in medicine (combining computer tomography (CT) and NMR data to obtain more complete information about the patient, monitoring tumor growth, treatment verification, comparison of the patient's data with anatomical atlases), in cartography (map updating), and in computer vision (target localization, automatic quality control), to name a few.

➤ **IMAGE COMPRESSION**

The objective of image compression is to reduce irrelevance and redundancy of the image data in order to be able to store or transmit data in an efficient form. Image compression may be lossy or lossless. Lossless compression is preferred for archival purposes and often for medical imaging, technical drawings, clip art, or comics. This is because lossy compression methods, especially when used at low bit rates, introduce compression artifacts. Lossy methods are especially suitable for natural images such as photographs in applications where minor (sometimes imperceptible) loss of fidelity is acceptable to achieve a substantial reduction in bit rate. The lossy compression that produces imperceptible differences may be called visually lossless.

➤ **IMAGE SYNTHESIS**

Image synthesis operations create images from other images or non-image data. Image synthesis operations generally create images that are either physically impossible or impractical to acquire.

1.1.2.4 APPLICATIONS OF IMAGE PROCESSING

Image processing has an enormous range of applications; almost every area of science and technology can make use of image processing methods. Here is a short list just to give some indication of the range of image processing applications.

➤ **MEDICINE**

Inspection and interpretation of images obtained from X-rays, MRI or CAT scans, analysis of cell images, of chromosome karyotypes. In medical applications, one is concerned with processing of chest X-rays, cineangiograms, projection images of trans axial tomography and other medical images that occur in radiology, nuclear magnetic resonance (NMR) and ultrasonic scanning. These images may be used for patient screening and monitoring or for detection of tumors' or other disease in patients.

➤ **AGRICULTURE**

Satellite/aerial views of land, for example to determine how much land is being used for different purposes, or to investigate the suitability of different regions for different crops, inspection of fruit and vegetables distinguishing good and fresh produce from old.

➤ **DOCUMENT PROCESSING**

It is used in scanning, and transmission for converting paper documents to a digital image form, compressing the image, and storing it on magnetic tape. It is also used in document reading for automatically detecting and recognizing printed characteristics.

➤ **RADAR IMAGING SYSTEM**

Radar and sonar images are used for detection and recognition of various types of targets or in guidance and maneuvering of aircraft or missile systems.

➤ **DEFENSE/INTELLIGENCE**

It is used in reconnaissance photo-interpretation for automatic interpretation of earth satellite imagery to look for sensitive targets or military threats and target acquisition and guidance for recognizing and tracking targets in real-time smart-bomb and missile-guidance systems.

1.2 OBJECTIVE

The motivations of this system are: (i) saving time for patients and (ii) to lower the workload of doctors by screening out the easy case. Another motivation for our project is to reduce human errors

1.3 EXISTING SYSTEM

- There are different types of medical imaging tools are available to detecting different types of abnormalities such as X-rays, Computed Tomography (CT), Magnetic Resonance Imaging (MRI), Ultrasound etc.
- X-rays and CT are most frequently used in fracture diagnosis because it is the fastest and easiest way for the doctors to study the injuries of bones and joints. Doctors usually uses x-ray images to determine whether a fracture exists, and the location of the fracture. The database is DICOM images
- In modern hospitals, medical images are stored in the standard DICOM (Digital Imaging and Communications in Medicine) format which includes text into the images. Any attempt to retrieve and display these images must go through PACS (Picture Archives and Communication System) hardware.

1.3.1 DISADVANTAGES OF EXISTING SYSTEM

- Depending on the human experts alone for such a critical matter can cause intolerable errors.

1.4 LITERATURE SURVEY

[1] Shubhangi D.C, Raghavendra S. Chinchansoor, P.S Hiremath, Edge Detection of Femur Bones in X-ray images – A Comparative study of Edge Detectors, Department of Computer Science, Poojya Doddappa Appa College of Engineering, Gulbarga – 585103 India, Volume 42-No.2, March 2012.

In this paper, we have examined the performance of Laplace operator in comparison with other edge detection methods in the literature, namely, Roberts, Sobel, Prewitt, and Canny's operators, which are applied to the X-ray images of femur bones. From the experimental results, it is observed that the Laplace operator gives better edge detection results than the other methods in the investigation of X-ray images of femur bones, which has significance to medical and forensic experts.

[2] Mahmoud Al-Ayyoub, Iamail Hmeidi, Haya Rababaha, Detecting Hand Bone Fractures in X-Ray Images, Jordan University of Science and Technology Irbid, Jordan, Volume 4. No.3, September 2013.

In this paper, the aim is to propose an efficient system for a quick and accurate diagnosis of hand bone fractures based on the information gained from the x-ray images. The general framework of the proposed system is as follows. It starts by taking a set of labeled x-ray hand images that contain normal as well as fractured hands and enhance them by applying some filtering algorithms to remove the noise from them. Then, it detects the edges in each image using edge detection methods. After that, it converts each image into a set of features using tools such the Wavelet and the Curvelet transforms. The next step is to build the classification algorithms based on the extracted features. Finally, in the testing phase, the performance and accuracy of the proposed system is evaluated.

[3] S.K. Mahndran, S. Santhosh BaBoo, An Enhanced Tibia Fracture Detection Tool Using Image Processing and Classification Fusion Techniques in X-Ray Images, Sankara College of Science and Commerce, Coimbatore, Tamil Nadu, India, Online ISSN: 0975-4172 &Print ISSN: 0975-4350, Volume 11 Issue 14 Version 1.0 August 2011.

Bone fractures are a common affliction and even in most developed countries the number of fractures associated with age-related bone loss and accidental fractures are increasing rapidly. From both orthopedic and radiologic point of view, the fully automatic detection and classification of fractures in long-bones is an important but difficult problem. The present work focuses on providing a solution to the automatic discovery of bone fracture in leg long bones. For this purpose, several image processing techniques (for preprocessing, segmentation and feature extraction) were used. The extracted features were then given as input to a fusion-based classification system to detect the presence / absence of fracture(s) in an image. Several experiments were conducted to analyze the performance of the proposed fusion classifier-based detection system with respect to its efficiency in terms of correct detection and speed of the algorithm. The performance was compared with its traditional single classification system. Experimental results proved that the proposed amalgamation of techniques showed improved results in terms of accuracy in detecting fractures and the speed of detection also. In future, other features like shape are to be considered and its effect on detection rate is to be analyzed. Moreover, its applicability to other long bones, like hand, back bone can also be analyzed.

[4] S.K. Mahndran, S. Santhosh BaBoo, An Ensemble Systems for Automatic Fracture Detection, IACIT International Journal of Engineering and Technology, Vol.4, No. 1, February 2012.

The main focus of the present research work is to automatically detect fractures in long bones from plain diagnostic X-Rays using a series of sequential steps. Three classifiers, namely, Back Propagation Neural Networks, Support Vector Machine and Naïve Bayes were considered. Two feature categories, texture and shape, were collected from the X-Ray image. Totally 11 features were extracted from the image which are used to detect the fracture bones through training and testing of classifiers. From these three base classifiers, four fusion classifiers were proposed. Experimental results proved that the fusion of classifier is efficient for fracture detection and achieved maximum accuracy. The time complexity of the algorithms was also on par with the

industry requirements. One difficulty encountered with fusion classification is the detection of a classifier which produces the best result. This process could be automated in future and the computer aided diagnosis program can intelligently identify the best combination of classifier and feature to produce highest performance. The present research work considers only simple fractures and experimental results showed that the performance degrades with fractures parallel to the bone edge are not detected as well as those perpendicular to the bone edges. Future research can consider these challenges.

[5] Rashmi, Mukesh Kumar, and Rohini Saxena, Algorithm and Technique on Various Edge Detection: A Survey, Department of Electronics and Communication Engineering, SHIATS-Allahabad, UP. -India, Vol. 4, No. 3, June 2013.

In this paper we have studied and evaluate different edge detection techniques. We have seen that canny edge detector gives better result as compared to others with some positive points. It is less sensitive to noise, adaptive in nature, resolved the problem of streaking, provides good localization and detects sharper edges as compared to others. It is considered as optimal edge detection technique hence lot of work and improvement on this algorithm has been done and further improvements are possible in future as an improved canny algorithm can detect edges in color image without converting in gray image, improved canny algorithm for automatic extraction of moving object in the image guidance. It finds practical application in Runway Detection and Tracking for Unmanned Aerial Vehicle, in brain MRI image, cable insulation layer measurement, Real-time facial expression recognition, edge detection of river regime, Automatic Multiple Faces Tracking and Detection. Canny edge detection technique is used in license plate reorganization system which is an important part of intelligent traffic system (ITS), finds practical application in traffic management, public safety and military department. It also finds application in medical field as in ultrasound, x –rays etc.

[6] Mahmoud Al-Ayyoub, Duha Al-Zghool, Determining the Type of Long Bone Fractures in X-Ray Images, Jordan University of Science & Technology, Irbid 22110, Jordan, E-ISSN: 2224-3402, Issue 8, Volume 10, August 2013.

In this paper, the proposed system uses image processing and machine learning techniques to accurately diagnose the existence and type of fracture in long bones. Specifically, it uses supervised learning in which the system classifies new instances based on a model built from a set of labeled examples (in this work, these are simply the x-ray images each with a normal/abnormal label) along with their distinguishing features (computed via image processing techniques). To be more specific, in the first step, a set of filtering algorithms is used to smooth the images and remove different types of noise such as: blurring, darkness, brightness, Poisson and Gaussian Noise. It then uses various tools to extract useful and distinguishing features based on: edge detection, corner detection, parallel & fracture lines, texture features, peak detection, etc. Due to the plethora of tools available for smoothing and noise removal and their high adaptability, significant effort is invested testing and tweaking them to find the ones that are most suitable for the problem at hand. The next step is to build our classification algorithms based on the extracted features to predict/classify fracture types. Finally, a testing phase is used to evaluate the performance and accuracy of the proposed process.

[7] Yuancheng — MIKE Luo and Ramani Duraiswami, Canny Edge Detection on NVIDIA CUDA, Computer Science & UMIACS, University of Maryland, College Park

In this paper, they demonstrated a version of the complete Canny edge detector under CUDA, including all stages of the algorithms. A significant speedup against straight forward CPU functions, but a moderate improvement against multi-core multi-threaded CPU functions taking advantage of special instructions was seen. The implementation speed is dominated by the hysteresis step (which was not implemented in previous GPU versions). If this postprocessing step is not needed the algorithm can be much faster (by a factor of four). It should emphasize that the algorithms used here could be made more efficient, and further speedups should be possible using more sophisticated component data parallel algorithms. Its experiences show that using CUDA one can move complex image processing algorithms to the GPU

[8] Zolertine, Habibollah Haron, Mohammed Rafiq Abdul Kadir, Comparison of Canny and Sobel Edge Detection in MRI Images, University Technology Malaysia.

From this paper, we can see that Canny method can produce equally good edge with the smooth continuous pixels and thin edge. Sobel edge detection method cannot produce smooth and thin edge compared to canny method. But same like other method, Sobel and Canny methods also very sensitive to the noise pixels. Sometime all the noisy image cannot be filtered perfectly. Unremoved noisy pixels will affect the result of edge detection. From our analysis, we have shown that between Sobel and Canny edge detection algorithms, response given by Canny edge detection was better than result of Sobel detector used in these MRI images.

1.5 PROPOSED SYSTEM

The X-ray/CT images are obtained from the hospital that contains normal as well as fractured bones images. In the first step, applying preprocessing techniques such as RGB to grayscale conversion and enhance them by using filtering algorithm to remove the noise from the image. Then it detects the edges in images using edge detection methods and segmented the image. After segmentation, it converts each image into a set of features by using some feature extraction technique. Then we build the classification algorithm based on extracted features. Finally, the performance and accuracy of the proposed system are evaluated

1.5.1 PROPOSED SYSTEM BLOCK DIAGRAM

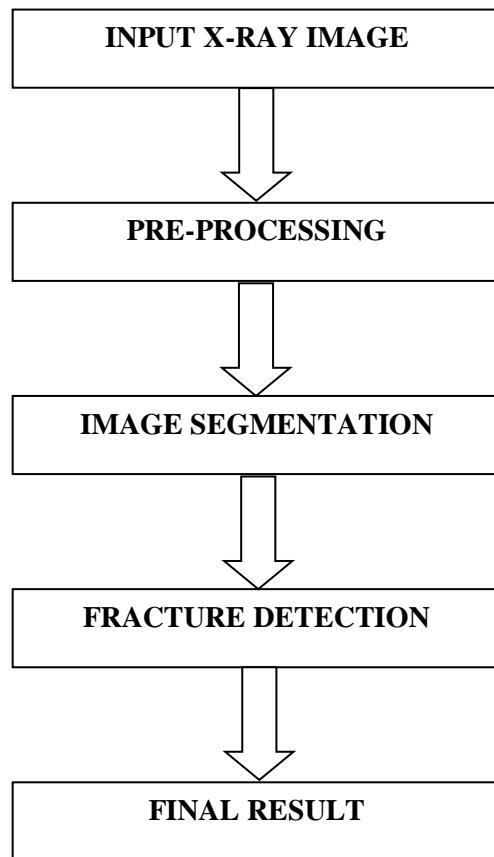


FIG 1.3: DETAILED BLOCK DIAGRAM OF PROPOSED SYSTEM

PROPOSED TECHNIQUE

- PREPROCESSING
- SEGMENTATION
- FRACTURE DETECTION

1.5.2 PROPOSED SYSTEM ADVANTAGES

- saving time for patients It is not sensitive to noise.

- to lower the workload of doctors by screening out the easy case

CHAPTER 2

PROJECT DESCRIPTION

2.1 INTRODUCTION

Bone fracture is common problem even in most developed countries and the number of fractures is increasing rapidly. Bone fracture can occur due to a simple accident or different types of diseases. So, quick and accurate diagnosis can be crucial to the success of any prescribed treatment. Depending on the human experts alone for such a critical matter have cause intolerable errors. Hence, the idea of automatic diagnosis procedure has always been an appealing one. The main goal of this project is to detect the lower leg bone fracture from X-Ray images using MATLAB software. The lower leg bone is the second largest bone of the body. It is made up of two bones, the tibia and fibula. The fibula bone is smaller and thinner than the tibia. However, the tibia fracture is most commonly occurring due to it carries a significant portion of the body weight. Among the four modalities (X-ray, CT, MRI, Ultrasound), X-ray diagnosis is commonly used for bone fracture detection due to their low cost, high speed and wide availability. Although CT and MRI images gives better quality images for body organs than X-ray images, the latter are faster cheaper, enjoy wider availability and are easier to use few limitations. Moreover, the level of quality of X-ray images is enough for the purpose of bone fracture detection.

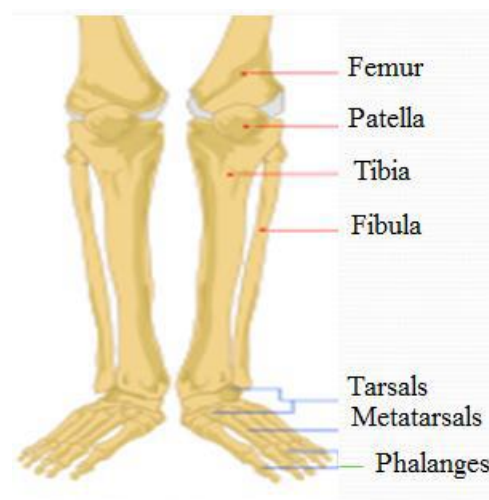


Figure 2.1. Structure of Lower Leg Bone

2.2 GENERAL

There are different types of noise such as poison, Gaussian, Salt & pepper, etc. Gaussian noise is the most common types of noise that can be found in X-ray images. This type of noise is generally caused by the sensor and circuitry of a scanner or digital camera. So, the system choses to use Gaussian filter to reduce the noise while preserving the edge and smooth of the image

2.2.1 PREPROCESSING:

This stage consists of the procedures that enhance the features of an input X-ray image so that the result image improves the performance of the subsequent stages of the proposed system. In this work, the main procedures for image enhancement are noise removal, adjusting image brightness and color adjustment. Noise can be defined as unwanted pixel that affects the quality of the image. The Gaussian smoothing filter is a very good filter for removing noise draw from a normal distribution. A Gaussian filter is parameterized by σ , and the relationship between σ and the degree of smoothing is very simple. A large σ implies a wider Gaussian filter and greater smoothing. After filtering, this system is performed adjusting image brightness and color to distinct the desired object or bone shape from the image. Then, the adjusted image is converted into the gray scale image to speed up processing time and less computation.



(a) *Original image*



(b) *Brightness adjusted image*

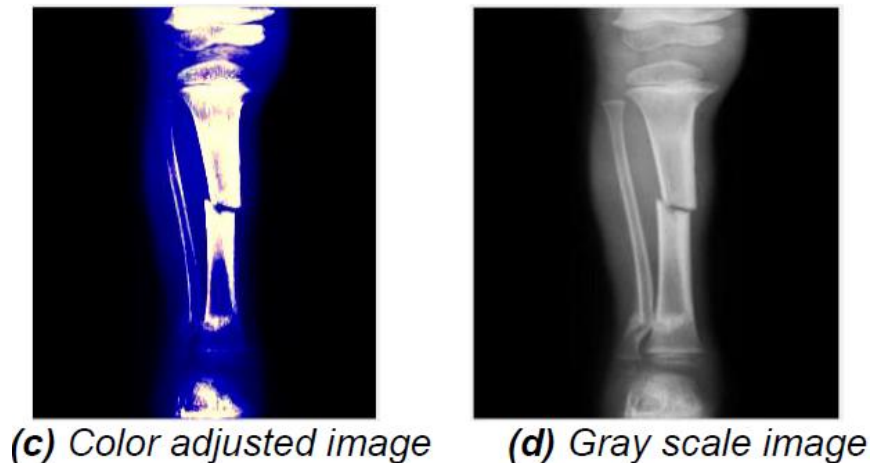


Figure 2.2. Results of Image preprocessing

2.3 EDGE DETECTION TECHNIQUES

In image processing especially in computer vision, the edge detection treats the localization of important variations of a gray level image and the detection of the physical and geometrical properties of objects of the scene. It is a fundamental process detects and outlines of an object and boundaries among objects and the background in the image. Edge detection is the most familiar approach for detecting significant discontinuities in intensity values. Edges are local changes in the image intensity. Edges typically occur on the boundary between two regions. The main features can be extracted from the edges of an image. Edge detection has major feature for image analysis.

These features are used by advanced computer vision algorithms. Edge detection is used for object detection which serves various applications like medical image processing, biometrics etc. Edge detection is an active area of research as it facilitates higher level image analysis. There are three different types of discontinuities in the grey level like point, line and edges. Spatial masks can be used to detect all the three types of discontinuities in an image. There are many edge detection techniques in the literature for image segmentation. The most commonly used discontinuity-based edge detection techniques are reviewed in this section.

Those techniques are

- Roberts edge detection
- Sobel Edge Detection
- Prewitt edge detection
- Canny Edge Detection
- LoG edge detection

2.3.1 Roberts Edge Detection

The Roberts edge detection is introduced by Lawrence Roberts (1965). It performs a simple, quick to compute, 2-D spatial gradient measurement on an image. This method emphasizes regions of high spatial frequency which often correspond to edges. The input to the operator is a grayscale image the same as to the output is the most common usage for this technique. Pixel values in every point in the output represent the estimated complete magnitude of the spatial gradient of the input image at that point.

-1	0
0	+1

G_x

0	-1
+1	0

G_y

2.3.2 Sobel Edge Detection

The Sobel edge detection method is introduced by Sobel in 1970. The Sobel method of edge detection for image segmentation finds edges using the Sobel approximation to the derivative. It precedes the edges at those points where the gradient is highest. The Sobel technique performs a 2-D spatial gradient quantity on an image and so highlights regions of high spatial frequency that correspond to edges. In general, it is used to find the estimated absolute gradient magnitude at each point in n input grayscale image. In conjecture at least the operator consists of a pair of 3x3 complication kernels as given away in under table. One kernel is simply the other rotated by 90o. This is very alike to the Roberts Cross operator.

-1	-2	-1
0	0	0
+1	+2	+1

G_x

-1	0	-1
-2	0	+2
-1	0	+1

G_y

2.3.3 Prewitt Edge Detection

The Prewitt edge detection is proposed by Prewitt in 1970. To estimate the magnitude and orientation of an edge Prewitt is a correct way. Even though different gradient edge detection wants a quite time-consuming calculation to estimate the direction from the magnitudes in the x and y-directions, the compass edge detection obtains the direction directly from the kernel with the highest response. It is limited to 8 possible directions; however, knowledge shows that most direct direction estimates are not much more perfect. This gradient based edge detector is estimated in the 3x3 neighborhood for eight directions. All the eight convolution masks are calculated. One complication mask is then selected, namely with the purpose of the largest module.

-1	-1	-1
0	0	0
+1	+1	+1

G_x

-1	0	+1
-1	0	+1
-1	0	+1

G_y

Prewitt detection is slightly simpler to implement computationally than the Sobel detection, but it tends to produce somewhat noisier results.

2.3.4 Canny Edge Detection

In industry, the Canny edge detection technique is one of the standard edge detections techniques. It was first created by John Canny for his Master's thesis at MIT in 1983, and still out performs many of the newer algorithms that have been developed. To find edges by separating noise from the image before find edges of image the Canny is a very important method. Canny method is a better method without disturbing the features of the edges in the image afterwards it applying the tendency to find the edges and the serious value for threshold.

The algorithmic steps are as follows:

- Convolve image $f(r, c)$ with a Gaussian function to get smooth image $f^{\wedge}(r, c)$.
 $f^{\wedge}(r, c) = f(r, c) * G(r, c, 6)$
- Apply first difference gradient operator to compute edge strength then edge magnitude and direction are obtained as before.
- Apply non-maximal or critical suppression to the gradient magnitude.
- Apply threshold to the non-maximal suppression image.

Unlike Roberts and Sobel, the Canny operation is not very susceptible to noise. If the Canny detector worked well it would be superior

2.3.5 LoG edge detection

The Laplacian of Gaussian (LoG) was proposed by Marr (1982). The LoG of an image $f(x, y)$ is a second order derivative defined as,

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

It has two effects, it will smooth the image and it computes the Laplacian, which yields a double edge image. Locating edges then consists of finding the zero crossings between the double edges. The digital implementation of the Laplacian function is usually made through the mask below,

0	-1	0
-1	4	-1
0	-1	0

G_x

-1	-1	-1
-1	8	-1
-1	-1	-1

G_y

The Laplacian is generally used to found whether a pixel is on the dark or light side of an edge.

2.3.6 RESULT

This section presents the relative performance of various edge detection techniques such as Roberts edge detector, Sobel Edge Detector, Prewitt edge detector, Kirsch, Robinson, Marr-Hildreth edge detector, LoG edge detector and Canny Edge Detector.

The edge detection techniques were implemented using MATLAB R2013a). The objective is to produce a clean edge map by extracting the principal edge features of the image. The original image and the image obtained by using different edge detection techniques are given in figure.

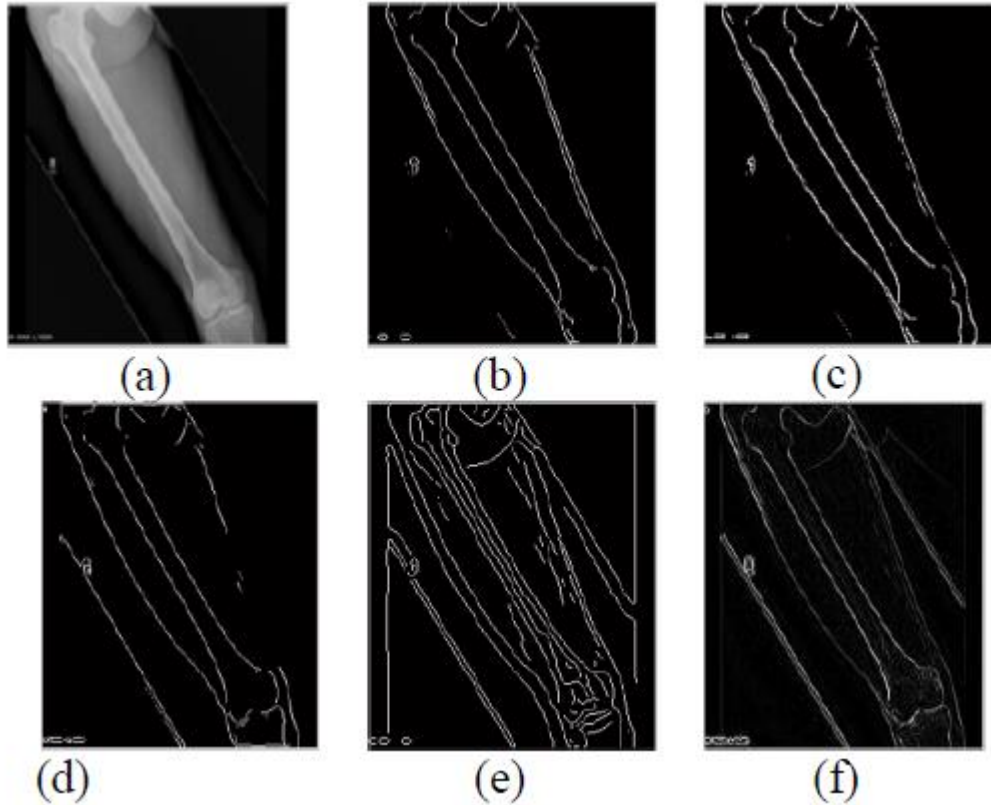


Fig. 2.3: (a) Original X-Ray input image and corresponding resultant edge detected images by using (b) Roberts, (c) Sobel, (d) Prewitt, (e) Canny, and (f) Laplace second order difference operators.

Roberts, Sobel and Prewitt results actually deviated from the others. LoG and Canny produce almost same edge map. It is observed from the figure, Canny result is superior by far to the other results.

2.4 Hough transform

2.4.1 Introduction

- The Hough transform (HT) can be used to detect lines, circles or other parametric curves.
- It was introduced in 1962 (Hough 1962) and first used to find lines in images a decade later (Duda 1972).
- The goal is to find the location of lines in images.
- This problem could be solved by e.g. Morphology and a linear structuring element, or by correlation.
- Then we would need to handle rotation, zoom, distortions etc.
- Hough transform can detect lines, circles and other structures if their parametric equation is known.
- It can give robust detection under noise and partial occlusion. It can give robust detection under noise and partial occlusion.

An image with linear structures:

- Borders between the regions are straight lines.
- These lines separate regions with different grey levels.
- Edge detection is often used as pre-processing to Hough transform.



Fig 2.4. linear Structured image before and after performing Edge Detection

2.4.2 Hough-transform – the input

- The input image must be a thresholder edge image.
- The magnitude results computed by the Sobel operator can be thresholded and used as input.

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Basic edge detection

- A thresholder edge image is the starting point for Hough transform.
- What does a canny filter produce?
- Approximation to the image gradient which is a vector quantity given by:

$$\nabla f(x, y) = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

Edge magnitude

- The gradient is a measure of how the function $f(x, y)$ changes as a function of changes in the arguments x and y .
- The gradient vector points in the direction of maximum change.
- The length of this vector indicates the size of the gradient:

$$|\nabla f| = \sqrt{G_x^2 + G_y^2}$$

G_x , G_y and the gradient operator

- Horizontal edges:

- Compute $g_x(x, y) = H_x * f(x, y)$

- Convolve with the horizontal filter kernel H_x

- Vertical edges: – Compute $g_y(x, y) = H_y * f(x, y)$

- Compute the gradient operator as:

$$g(x, y) = \sqrt{g_x^2(x, y) + g_y^2(x, y)} \quad \text{Gradient-magnitude (kant-styrke)}$$

$$\theta(x, y) = \tan^{-1}\left(\frac{g_y(x, y)}{g_x(x, y)}\right) \quad \text{Gradient-retning}$$

Edge direction

- The direction of this vector is also an important quantity.

- If $\alpha(x, y)$ is the direction of f in the point (x, y) then:

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

- Remember that $\alpha(x, y)$ will be the angle with respect to the x-axis.

- Remember also that the direction of an edge will be perpendicular to the gradient in any given point.

2.4.3 Input to Hough – thresholded edge image

Prior to applying Hough transform:

- Compute edge magnitude from input image.
- As always with edge detection, simple lowpass filtering can be applied first.
- Threshold the gradient magnitude image.

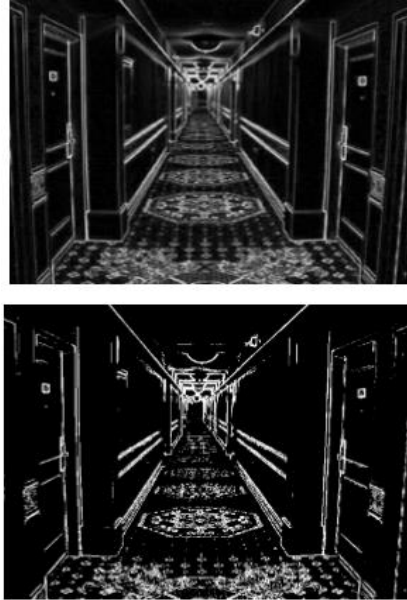


Fig.2.5. Edge Detected Image and Thresholded image

Hough-transform

- Assume that we have performed some edge detection, and a thresholding of the edge magnitude image.
- Thus, we have n pixels that may partially describe the boundary of some objects.
- We wish to find sets of pixels that make up straight lines.
- Regard a point (x_i, y_i) and a straight line $y_i = a x_i + b$
 - There are many lines passing through the point (x_i, y_i) .
 - Common to them is that they satisfy the equation for some set of parameters (a, b) .

2.4.4 Hough transform basic idea

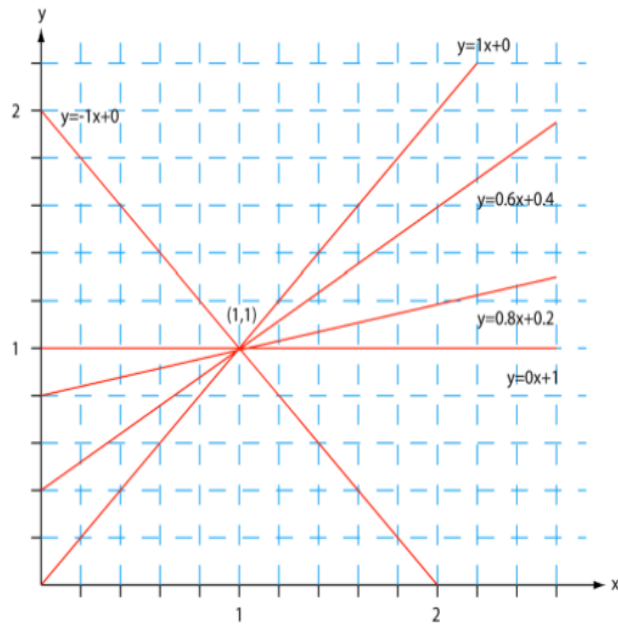


Fig.2.6. Plot of Different Lines passing through a Single point

- This equation can obviously be rewritten as follows:

$$b = -xa + y$$

- We now consider x and y as parameters and a and b as variables.
- This is a line in (a, b) space parameterized by x and y .
 - So, a single point in xy -space gives a line in (a, b) space.
- Another point (x, y) will give rise to another line in (a, b) space.

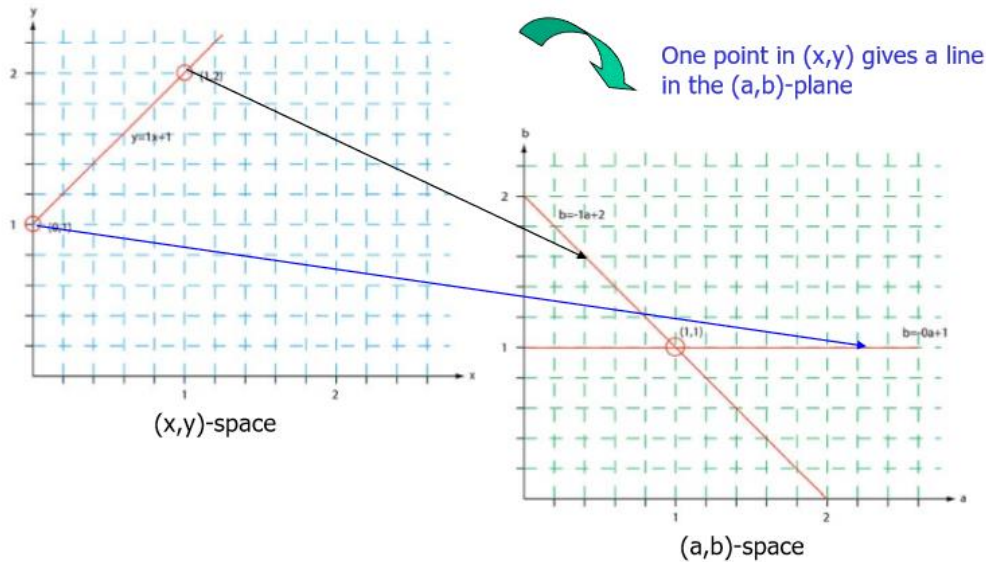


Fig.2.7. Plot Showing the transformation of a single point

- Two points (x, y) and (z, k) define a line in the (x, y) plane.
- These two points give rise to two different lines in (a, b) space.
- In (a, b) space these lines will intersect in a point (a^1, b^1) where a^1 is the rise and b^1 the intersect of the line defined by (x, y) and (z, k) in (x, y) space.
- The fact is that all points on the line defined by (x, y) and (z, k) in (x, y) space will parameterize lines that intersect in (a^1, b^1) in (a, b) space.
- Points that lie on a line will form a “cluster of crossings” in the (a, b) space.

2.4.4.1 Hough transform – algorithm

- Quantize the parameter space (a, b) , that is, divide it into cells.
- This quantized space is often referred to as the accumulator cells.
- In the below figure a_{\min} is the minimal value of a etc.
- Count the number of times a line intersects a given cell.
 - For each point (x, y) with value 1 in the binary image, find the values of (a, b) in the range

$[[a_{\min}, a_{\max}], [b_{\min}, b_{\max}]]$ defining the line corresponding to this point.

- Increase the value of the accumulator for these $[a^1, b^1]$ point.
- Then proceed with the next point in the image.
- Cells receiving a minimum number of “votes” are assumed to correspond to lines in (x, y) space.
- Lines can be found as peaks in this accumulator space.

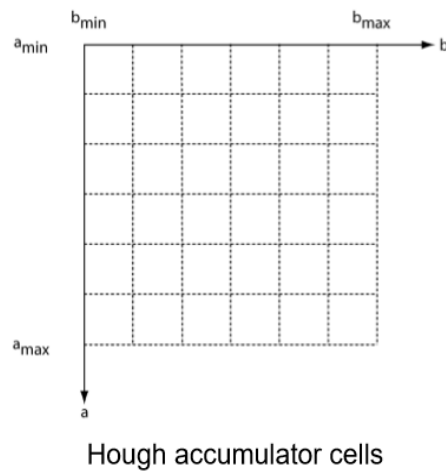


Fig.2.8. Hough Accumulator Cells

2.4.4.2 Hough transform – polar representation of lines

- In practical life we use the polar representation of lines:

$$x \cos \theta + y \sin \theta = \rho$$

- The polar (also called normal) representation of straight lines is

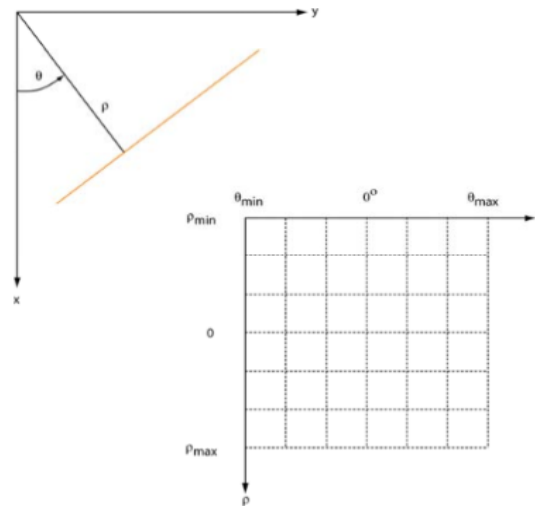
$$x \cos \theta + y \sin \theta = \rho$$

- Each point (x_i, y_i) in the xy -plane gives a sinusoid in the $\rho\theta$ - plane.
- M colinear point lying on the line

$$x \cos \theta + y \sin \theta = \rho$$

will give M curves that intersect at (ρ_i, θ_j) in the parameter plane.

- Local maxima give significant lines.



Polar representation of lines

Fig.2.9.1 Polar Representation of lines

- Each curve in the figure represents the family of lines that pass through a particular point (x_i, y_i) in the xy -plane.
- The intersection point (ρ^1, θ^1) corresponds to the lines that passes through two points (x_i, y_i) and (x_j, y_j) .
- A horizontal line will have $\theta=0$ and ρ equal to the intercept with the y -axis.
- A vertical line will have $\theta=90$ and ρ equal to the intercept with the x -axis.

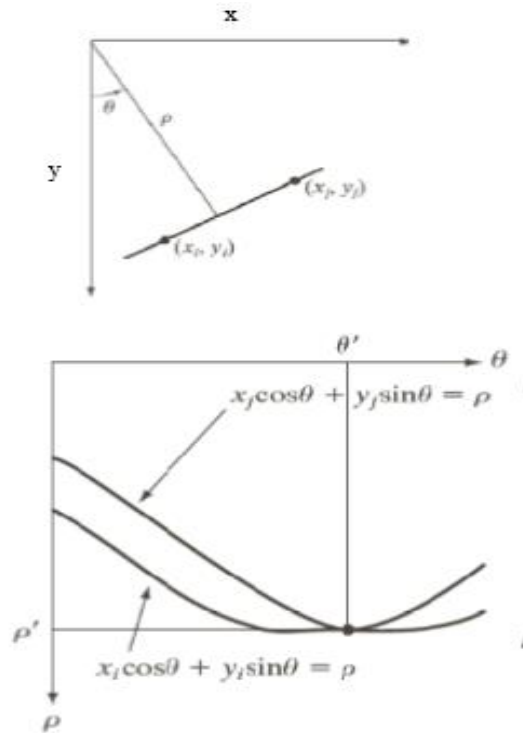


Fig.2.9.2. Polar Representation of lines

2.4.4.3 Hough transform - algorithm using polar representation of lines

- Partition the $\rho\theta$ -plane into accumulator cells $A[\rho, \theta]$, $\rho \in [\rho_{\min}, \rho_{\max}]$; $\theta \in [\theta_{\min}, \theta_{\max}]$
- The range of θ is $\pm 90^\circ$
 - Horizontal lines have $\theta=0^\circ$, $\rho \geq 0$,
 - Vertical lines have $\theta=90^\circ$, $\rho \geq 0$
- The range of ρ is $\pm N\sqrt{2}$ if the image is of size $N \times N$.
- The discretization of θ and ρ must happen with values $\delta\theta$ and $\delta\rho$ giving acceptable precision and sizes of the parameter space.
- The cell (i, j) corresponds to the square associated with parameter values (θ_j, ρ_i) .
- Initialize all cells with value 0.

- For each foreground point (x_k, y_k) in the thresholded edge image
 - Let θ_j equal all the possible θ -values.
 - Solve for ρ using $\rho = x \cos \theta_j + y \sin \theta_j$.
 - Round ρ to the closest cell value, ρ_q .
 - Increment $A(i, q)$ if the θ_j results in ρ_q .
- After this procedure, $A(i, j) = P$ means that P points in the xy space lie on the line

$$\rho_j = x \cos \theta_j + y \sin \theta_j.$$

- Find line co-ordinates where $A(I, j)$ is above a suitable threshold value.

• **Example:**

Natural scene and result of canny edge detection:



Fig.2.10. Original and Sobel Edge Detected Image

Natural scene and result of Sobel edge detection followed by thresholding:



Fig.2.11. Original and Thresholded Image

Original image and 20 most prominent lines:



Fig.2.12. Original and Hough Transformed Image

2.4.5 Hough transform

➤ Advantages

- Conceptually simple.
- Easy implementation Easy implementation.
- Handles missing and occluded data very gracefully.
- Can be adapted to many types of forms, not just lines.

➤ Disadvantages

- Computationally complex for objects with many parameters.
- Looks for only one single type of object.
- Can be “fooled” by “apparent lines”.
- The length and the position of a line segment cannot be determined.
- Co-linear line segments cannot be separated.

2.7 METHODOLOGIES

2.7.1 MODULE NAMES

- Input Image conversion
- Preprocessing
- Segmentation
- Fracture detection.

MODULE 1

RGB TO GRAY CONVERSION

The original image (X-ray) is in an uncompressed format and that the pixel values are within $[0, 255]$, and denote the numbers of rows and columns as N_1 and N_2 and the pixel number as $(N=N_1 \times N_2)$. Since it was in RGB (color) format, it was converted into grayscale using RGB to gray conversion process. Image Resizing process also done if needed.

MODULE 2

PREPROCESSING:

The original image is smoothed implementing with a Gaussian filter. The result is an image with less blur. It is intended to obtain the real edges of the image.

MODULE 3

SEGMENTATION:

The algorithmic steps are as follows:

- Convolve image $f(r, c)$ with a Gaussian function to get smooth image $f^{\wedge}(r, c)$.

$$f^{\wedge}(r, c) = f(r, c) * G(r, c, \sigma)$$

- Apply first difference gradient operator to compute edge strength then edge magnitude and direction are obtained as before.
- Apply non-maximal or critical suppression to the gradient magnitude.
- Apply threshold to the non-maximal suppression image.

Unlike Roberts and Sobel, the Canny operation is not very susceptible to noise. If the Canny detector worked well it would be superior.

MODULE 4

FRACTURE DETECTION:

The last stage of this system is fracture detection it is performed by the procedures. First, the useful features such as straight lines can be extracted from the image. And then, these features are used to detect fracture or non-fracture image. After enhancing and segmentation the input image, the process is extracted the features in binary image by using Hough transform. The Hough transform is a feature extraction technique it is concerned with the identification of straight lines, shapes, curves in a given image. It takes a binary image as an input.

CHAPTER 3

SOFTWARE SPECIFICATION

3.1 MATLAB INTRODUCTION

MATLAB (matrix laboratory) is a numerical computing environment and fourth-generation programming language. Developed by Math Works, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, Java, and Fortran.

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing capabilities. An additional package, Simulink, adds graphical multi-domain simulation and Model-Based Design for dynamic and embedded systems.

In 2004, MATLAB had around one million users across industry and academia. MATLAB users come from various backgrounds of engineering, science, and economics. MATLAB is widely used in academic and research institutions as well as industrial enterprises.

MATLAB was first adopted by researchers and practitioners in control engineering, Little's specialty, but quickly spread to many other domains. It is now also used in education, in particular the teaching of linear algebra and numerical analysis, and is popular amongst scientists involved in image processing. The MATLAB application is built around the MATLAB language. The simplest way to execute MATLAB code is to type it in the Command Window, which is one of the elements of the MATLAB Desktop. When code is entered in the Command Window, MATLAB can be used as an interactive mathematical shell. Sequences of commands can be saved in a text file, typically using the MATLAB Editor, as a script or encapsulated into a function, extending the commands available.

MATLAB provides a number of features for documenting and sharing your work. You can integrate your MATLAB code with other languages and applications, and distribute your MATLAB algorithms and applications.

3.2 FEATURES OF MATLAB

- High-level language for technical computing.
- Development environment for managing code, files, and data.
- Interactive tools for iterative exploration, design, and problem solving.
- Mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, and numerical integration.
- 2-D and 3-D graphics functions for visualizing data.
- Tools for building custom graphical user interfaces.
- Functions for integrating MATLAB based algorithms with external applications and languages, such as C, C++, Fortran, Java™, COM, and Microsoft Excel.

MATLAB is used in vast area, including signal and image processing, communications, control design, test and measurement, financial modeling and analysis, and computational. Add-on toolboxes (collections of special-purpose MATLAB functions) extend the MATLAB environment to solve particular classes of problems in these application areas.

MATLAB can be used on personal computers and powerful server systems, including the Cheaha compute cluster. With the addition of the Parallel Computing Toolbox, the language can be extended with parallel implementations for common computational functions, including for-loop unrolling. Additionally this toolbox supports offloading computationally intensive workloads to Cheaha the campus compute cluster. MATLAB is one of a few languages in which each variable is a matrix (broadly construed) and "knows" how big it is. Moreover, the fundamental operators (e.g. addition, multiplication) are programmed to deal with matrices when required. And the MATLAB environment handles much of the bothersome housekeeping that makes all this possible. Since so many of the procedures required for Macro-Investment Analysis involves matrices, MATLAB proves to be an extremely efficient language for both communication and implementation.

3.2.1 INTERFACING WITH OTHER LANGUAGES

MATLAB can call functions and subroutines written in the C programming language or FORTRAN. A wrapper function is created allowing MATLAB data types to be passed and returned. The dynamically loadable object files created by compiling such functions are termed "MEX-files" (for **M**ATLAB **e**xecutable).

Libraries written in Java, ActiveX or .NET can be directly called from MATLAB and many MATLAB libraries (for example XML or SQL support) are implemented as wrappers around Java or ActiveX libraries. Calling MATLAB from Java is more complicated, but can be done with MATLAB extension, which is sold separately by Math Works, or using an undocumented mechanism called JMI (Java-to-Mat lab Interface), which should not be confused with the unrelated Java that is also called JMI.

As alternatives to the MuPAD based Symbolic Math Toolbox available from Math Works, MATLAB can be connected to Maple or Mathematica.

Libraries also exist to import and export MathML.

Development Environment

- Startup Accelerator for faster MATLAB startup on Windows, especially on Windows XP, and for network installations.
- Spreadsheet Import Tool that provides more options for selecting and loading mixed textual and numeric data.
- Readability and navigation improvements to warning and error messages in the MATLAB command window.
- Automatic variable and function renaming in the MATLAB Editor.

Developing Algorithms and Applications

MATLAB provides a high-level language and development tools that let you quickly develop and analyze your algorithms and applications.

The MATLAB Language

The MATLAB language supports the vector and matrix operations that are fundamental to engineering and scientific problems. It enables fast development and execution. With the MATLAB language, you can program and develop algorithms faster than with traditional languages because you do not need to perform low-level administrative tasks, such as declaring variables, specifying data types, and allocating memory. In many cases, MATLAB eliminates the need for ‘for’ loops. As a result, one line of MATLAB code can often replace several lines of C or C++ code.

At the same time, MATLAB provides all the features of a traditional programming language, including arithmetic operators, flow control, data structures, data types, object-oriented programming (OOP), and debugging features.

MATLAB lets you execute commands or groups of commands one at a time, without compiling and linking, enabling you to quickly iterate to the optimal solution. For fast execution of heavy matrix and vector computations, MATLAB uses processor-optimized libraries. For general-purpose scalar computations, MATLAB generates machine-code instructions using its JIT (Just-In-Time) compilation technology.

This technology, which is available on most platforms, provides execution speeds that rival those of traditional programming languages.

MATLAB Editor

Provides standard editing and debugging features, such as setting breakpoints and single stepping

Code Analyzer

Checks your code for problems and recommends modifications to maximize performance and maintainability

MATLAB Profiler

Records the time spent executing each line of code

Directory Reports

Scan all the files in a directory and report on code efficiency, file differences, file dependencies, and code coverage.

Designing Graphical User Interfaces

By using the interactive tool GUIDE (Graphical User Interface Development Environment) to layout, design, and edit user interfaces. GUIDE lets you include list boxes, pull-down menus, push buttons, radio buttons, and sliders, as well as MATLAB plots and Microsoft ActiveX® controls. Alternatively, you can create GUIs programmatically using MATLAB functions.

3.2.2 ANALYZING AND ACCESSING DATA

MATLAB supports the entire data analysis process, from acquiring data from external devices and databases, through preprocessing, visualization, and numerical analysis, to producing presentation-quality output.

Data Analysis

MATLAB provides interactive tools and command-line functions for data analysis operations, including:

- Interpolating and decimating
- Extracting sections of data, scaling, and averaging
- Thresholding and smoothing
- Correlation, Fourier analysis, and filtering
- 1-D peak, valley, and zero finding
- Basic statistics and curve fitting
- Matrix analysis

Data Access

MATLAB is an efficient platform for accessing data from files, other applications, databases, and external devices. You can read data from popular file formats, such as Microsoft Excel; ASCII text or binary files; image, sound, and video files; and scientific files, such as HDF and HDF5. Low-level binary file I/O functions let you work with data files in any format. Additional functions let you read data from Web pages and XML.

Visualizing Data

All the graphics features that are required to visualize engineering and scientific data are available in MATLAB. These include 2-D and 3-D plotting functions, 3-D volume visualization functions, tools for interactively creating plots, and the ability to export results to all popular graphics formats. You can customize plots by adding multiple axes; changing line colors and markers; adding annotation, Latex equations, and legends; and drawing shapes.

2-D Plotting

Visualizing vectors of data with 2-D plotting functions that create:

- Line, area, bar, and pie charts.
- Direction and velocity plots.
- Histograms.
- Polygons and surfaces.
- Scatter/bubble plots.
- Animations.

3-D Plotting and Volume Visualization

MATLAB provides functions for visualizing 2-D matrices, 3-D scalar, and 3-D vector data. You can use these functions to visualize and understand large, often complex, multidimensional data. Specifying plot characteristics, such as camera viewing angle, perspective, lighting effect, light source locations, and transparency.

3-D plotting functions include:

- Surface, contour, and mesh.
- Image plots.
- Cone, slice, stream, and iso-surface.

3.2.3 PERFORMING NUMERIC COMPUTATION

MATLAB contains mathematical, statistical, and engineering functions to support all common engineering and science operations. These functions, developed by experts in mathematics, are the foundation of the MATLAB language. The core math functions use the LAPACK and BLAS linear algebra subroutine libraries and the FFTW Discrete Fourier Transform library. Because these processor-dependent libraries are optimized to the different platforms that MATLAB supports, they execute faster than the equivalent C or C++ code.

MATLAB provides the following types of functions for performing mathematical operations and analyzing data:

- Matrix manipulation and linear algebra.
- Polynomials and interpolation.
- Fourier analysis and filtering.
- Data analysis and statistics.
- Optimization and numerical integration.
- Ordinary differential equations (ODEs).
- Partial differential equations (PDEs).
- Sparse matrix operations.

MATLAB can perform arithmetic on a wide range of data types, including doubles, singles, and integers.

CHAPTER 4

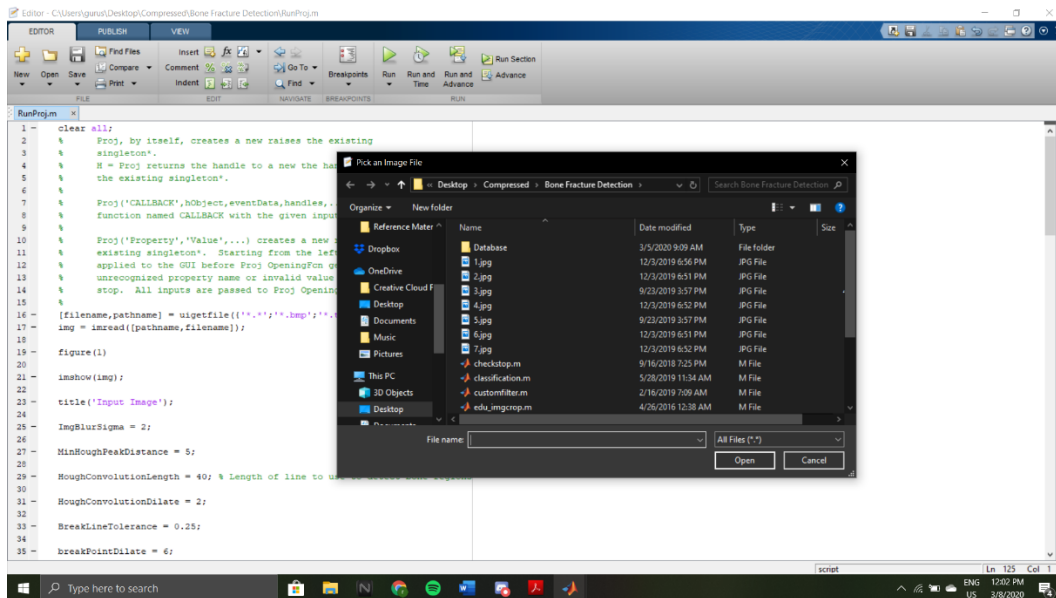
RESULTS

4.1 IMPLEMENTATION

MATLAB is a program that was originally designed to simplify the implementation of numerical linear algebra routines. It has since grown into something much bigger, and it is used to implement numerical algorithms for a wide range of applications. The basic language used is very similar to standard linear algebra notation, but there are a few extensions that will likely cause you some problems at first.

4.2 SNAPSHOTS

SNAPSHOT TO SELECT AN X-RAY IMAGE



4.2.1. ORIGINAL X RAY IMAGE

Input Image

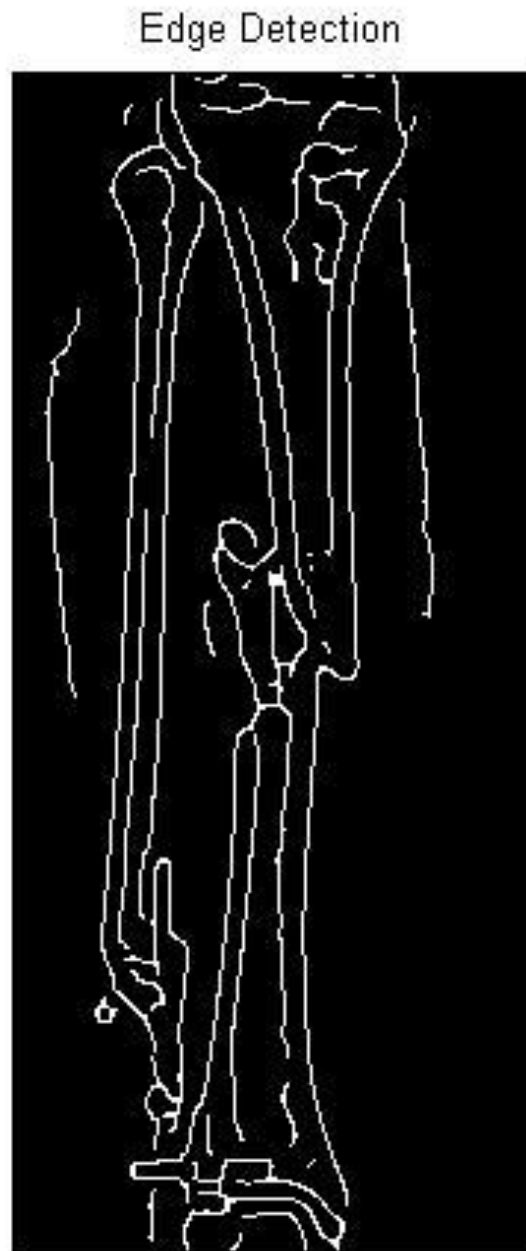


4.2.2. FILTERED IMAGE

Filtered image



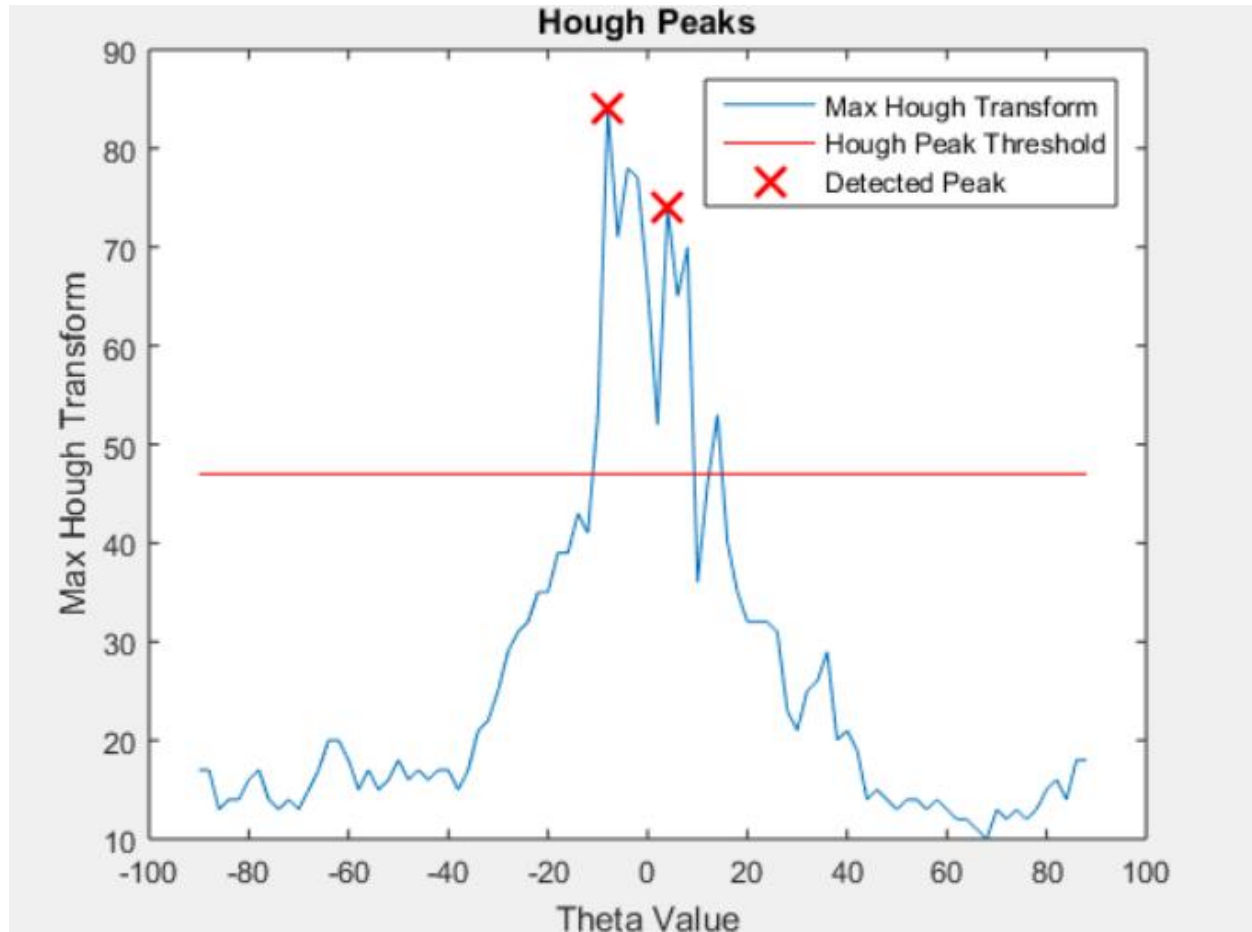
4.2.3. EDGE DETECTION IMAGE



4.2.4. HOUGH-LINES IMAGE

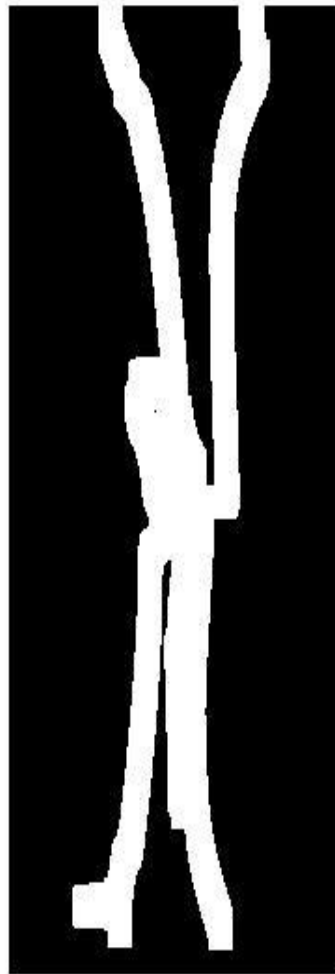


4.2.5. HOUGH-PEAKS IMAGE



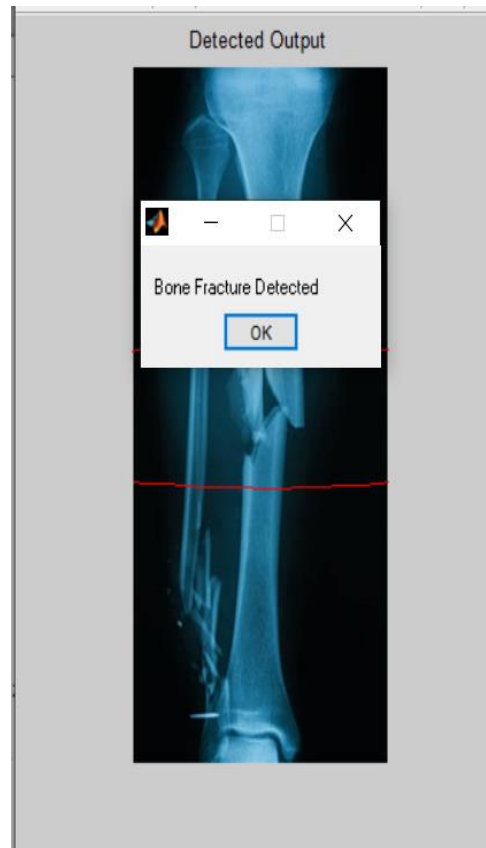
TWO PEAKS DETECTION INDICATES A FRACTURE PRESENT

4.2.6. HOUGH-THRESHOLDED IMAGE



Area where Break line Tolerance is Crossed

4.2.7. FINAL IMAGE



CHAPTER 5

APPLICATIONS

- Used for effective analysis of the leg fractures and provides a crystal-clear view of the Fracture.
- Provides accurate result.
- Saves the analysis time that doctors take to Identify the Fracture.
- Burden on the doctors can also be reduced.
- Chance for Human errors can be reduced using this method.

CHAPTER 6

CONCLUSION AND REFERENCES

6.1 CONCLUSION

This project presented the image processing technique to detect the bone fracture. The fully automatic detection of fractures in leg bone is an important but difficult problem. According to the test results, the system has been done to detect the bone fracture. A conclusion can be made that the performance of the detection method affected by the quality of the image. The better the image quality, the better the result system got. In future work, focusing on other works like detecting on smaller bone, ankle fractures, etc. may be considered.

6.2 REFERENCES

[1] Shubhangi D.C, Raghavendra S. Chinchansoor, P.S Hiremath, Edge Detection of Femur Bones in X-ray images – A Comparative study of Edge Detectors, Department of Computer Science, Poojya Doddappa Appa College of Engineering, Gulbarga – 585103 India, Volume 42-No.2, March 2012.

[2] Mahmoud Al-Ayyoub, Iamail Hmeidi, Haya Rababaha, Detecting Hand Bone Fractures in X-Ray Images, Jordan University of Science and Technology Irbid, Jordan, Volume 4. No.3, September 2013.

[3] S.K. Mahndran, S. Santhosh BaBoo, An Enhanced Tibia Fracture Detection Tool Using Image Processing and Classification Fusion Techniques in X-Ray Images, Sankara College of Science and Commerce, Coimbatore, Tamil Nadu, India, Online ISSN: 0975-4172 &Print ISSN: 0975-4350, Volume 11 Issue 14 Version 1.0 August 2011.

- [4] S.K. Mahndran, S. Santhosh BaBoo, An Ensemble Systems for Automatic Fracture Detection, IACIT International Journal of Engineering and Technology, Vol.4, No. 1, February 2012.
- [5] Rashmi, Mukesh Kumar, and Rohini Saxena, Algorithm and Technique on Various Edge Detection: A Survey, Department of Electronics and Communication Engineering, SHIATS-Allahabad, UP. -India, Vol. 4, No. 3, June 2013.
- [6] Mahmoud Al-Ayyoub, Duha Al-Zghool, Determining the Type of Long Bone Fractures in X-Ray Images, Jordan University of Science & Technology, Irbid 22110, Jordan, E-ISSN: 2224-3402, Issue 8, Volume 10, August 2013.
- [7] Yuancheng — MIKE|| Luo and Ramani Duraiswami, Canny Edge Detection on NVIDIA CUDA, Computer Science & UMIACS, University of Maryland, College Park
- [8] Zolqernine, Habibollah Haron, Mohammed Rafiq Abdul Kadir, Comparison of Canny and Sobel Edge Detection in MRI Images, University Technology Malaysia.

Detection of Leg Fracture in X-Ray Images using Image Processing

Siva Kumar A¹, Deva HarshithaBarre², Venkata Guru Saran Dasari³,

RavisaiTejaYerra⁴, ManoharEdavaka⁵

Dept of ECE, Anil Neerukonda Institute of Techonology and Sciences

¹ Sivakumar.ece@anits.edu.in

² harshi.deva@gmail.com

³ gurusarand@gmail.com

⁴ raviteja01234.yerra@gmail.com

⁵ eduvakamanohar@gmail.com

Abstract—Quickly creating innovations are developing each day in various fields, particularly in restorative condition. Notwithstanding, still some old strategies are very famous. X Rays are one of these systems for identification of bone cracks. By the way, here and there the span of breaks isn't huge and couldn't be recognized effectively. So, for the efficient recognition of the crack has become more important. This project plans to build up a sharp characterization framework that would be equipped for identifying and characterizing the bone cracks. The procedures of image processing such as image enhancement, image segmentation and feature extraction are used for fracture recognition system. This paper which we are proposing uses Canny edge detection method for segmentation. Canny edge detection method produces ideal information from the bone figure. The main aim of this study and research is to identify human lower leg bone fractures from X-Ray images. The proposed system has three steps, i.e., preprocessing, segmentation, and fracture detection. In feature extraction stage, we have used Hough transform technique for line detection in the image. Feature extraction is the main task of the system. The results from various experiments show that the proposed system is very accurate and efficient.

Keywords— X-Ray images, Image processing, Fracture detection

I. INTRODUCTION

Bone fracture is frequent occurred problem even in most developed countries and the number of fractured cases is increasing rapidly. Bone fracture can occur due to a small accident or different types of complications in the human body. So, quick and accurate diagnosis can be crucial to the success of any prescribed treatment. Depending on the human experts alone for such a critical matter have caused intolerable errors. Hence, the idea of automatic diagnosis practice has always been an interesting one. The main objective of this project is to detect the lower leg bone fracture from X-Ray images using MATLAB software. The lower leg bone i.e., the femur bone is the second largest bone of the body. It is made up of two bones, the tibia and fibula. The fibula bone is much smaller and thinner than the tibia bone. However, the tibia fracture occurs most commonly due to which it carries a major portion of the body weight. Among the resources we have i.e., X-ray, CT, MRI, and Ultrasound. X-ray analysis is commonly used for bone fracture recognition due to their low cost, high speed and wide accessibility. Although CT and MRI imagery gives us better superiority images for body organs than X-ray images, the latter are faster and cheaper, we can enjoy wider availability and are easier to use and have very few limitations. Moreover, the level of excellence of X-ray images is enough for the function of bone fracture detection. The motivation of this project is, saving the time for patients and to reduce the workload of doctors by screening out the uncomplicated cases. Another inspiration for our study and research is to reduce human errors because doctors in hospitals physically scrutinize a large number of X-ray images for fractures. Manual examination of the bone fractures is tedious and time-consuming process which needs a lot of attention. A tired radiologist has been found to overlook a fracture image among the healthy ones. Computer vision method can help to monitor X-ray images of doubtful cases and alarm the doctors.

II. PREVIOUS STUDY

Samuel FebriantoKurniawan, presented the Canny Edge Detection method to assist radiologists in recognizing fractured bones from different X-ray images [6]. They have observed that simulation results show that the classification requires to be enhanced on its performance and decrease the response point in time. This work has been made to a conclusion that the performance and accurateness of the recognition system is affected by the superiority of the image. Ms.SnehaDeshmukh, Ms.ShivaniZalte, has applied various image processing techniques to find a fracture in the bone [7]. The authors have compared the different edge detectors and have described the pros and cons of these detectors. It is justified that the Canny edge detection technique produce uniformly superior edges with smooth continuous pixels and thin edges. Sobel edge detection method cannot construct smooth and thin edges when compared to Canny edge detection method. But like other methods as we have, Sobel technique and Canny methods are also very susceptible to the noise pixels. Sometimes all the noisy images cannot be filtered completely with the above methods. Tai Peng Tian, Ying Chen, Wynne Hsu, has analyzed a process of computing neck-shaft angle for recognizing femur bone fracture in the human body [8]. This task is performed by the withdrawal of femur contour via a combination of algorithms, i.e., Canny edge detection method and Hough transform method for recognizing important straight line and curve characters.

III. PROPOSED SYSTEM

In this section, we will discuss about the detailed steps involved in obtaining the final result. The steps involved here are Pre-Processing i.e., filtering using gaussian function, Segmentation using canny edge detector and Fracture detection using Hough Transform.

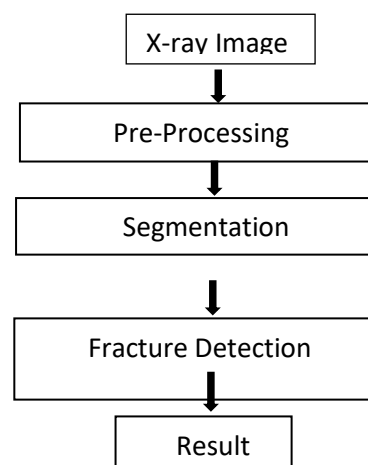


Figure 1 .System Flow Diagram

A. Preprocessing

In this Stage we perform certain processes on the image like converting an image into gray scale image and applying filtering techniques. The Main task of this stage is to make the image free from noise in order to obtain a perfect output. An X- ray image consists of gaussian noise generally. So, to remove this type of noise we use gaussian filter, but before that we need to gray scale the image. Now, this gray-scaled image is given as input to the gaussian filter. The gaussian filtering level varies with the value of σ . This Preprocessed image is now used in the next step i.e., segmentation. The following figures (b) and (c) gives the gray-scaled image and filtered image respectively applied on the Fig (a).

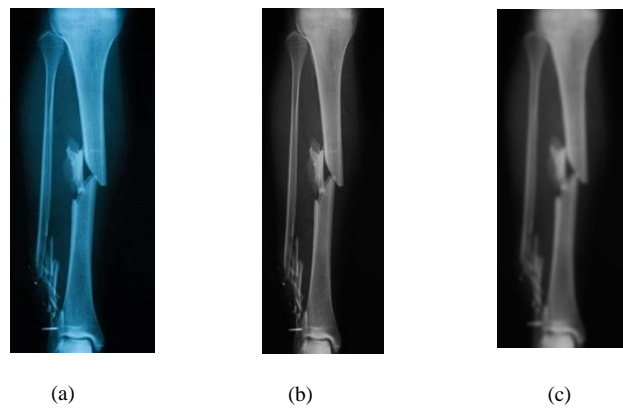


Figure 2. Results of Image pre-processing

B. Segmentation:

Image segmentation is a commonly used technique in digital image processing and analysis to partition an image into multiple parts or regions, often based on the characteristics of the pixels in the image. Edge detection is the most familiar approach for detecting significant discontinuities in intensity values. Edges are local changes in the image intensity. Edges typically occur on the boundary between two regions. The main features can be extracted from the edges of an image. Edge detection has major feature for image analysis. These features are used by advanced computer vision algorithms. Edge detection is used for object detection which serves various applications like medical image processing, biometrics etc. Edge detection is an active area of research as it facilitates higher level image analysis. There are three different types of discontinuities in the grey level like point, line and edges. Spatial masks can be used to detect all the three types of discontinuities in an image. There are many edge detection techniques in the literature for image segmentation.

Canny method is a better method without disturbing the features of the edges in the image afterwards it applying the tendency to find the edges and the serious value for threshold.

The algorithmic steps are as follows:

- Convolve image $f(r, c)$ with a Gaussian function to get smooth image $f^{\wedge}(r, c)$

$$f^{\wedge}(r, c) = f(r, c) * G(r, c, \sigma)$$
- Apply first difference gradient operator to compute edge strength then edge magnitude and direction are obtained as before.
- Apply non-maximal or critical suppression to the gradient magnitude.
- Apply threshold to the non-maximal suppression image.

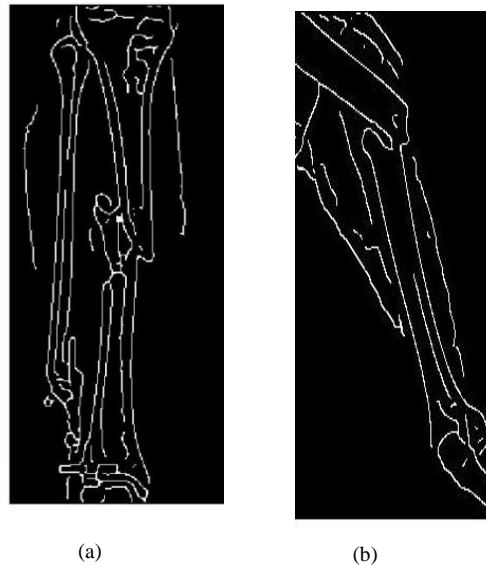


Figure 3. Result images of edge detector

C. Fracture Detection

The Fracture Detection part here, uses Hough Transform to find the area of the fracture in the image. The Hough Transform is generally used to find any sort of lines, circles or any structure if their Parametric equation is known. The Basic ideology of Hough Transform is to represent a point in (x, y) plane to a corresponding line or curve in Parameter Space. In general, we represent a line in polar form i.e., $x\cos\theta + y\sin\theta = \rho$. Every point in the (x, y) plane is transformed into (ρ, θ) .

Plane as a curve. A vertical line will have $\theta = 90$, a horizontal line will have $\theta = 0$, a line passing below the origin will have a positive rho and angle less than 180 and if the line is above the origin it will have a negative rho but the angle is less than 180. The (ρ, θ) plane is quantized and is divided into different cells called accumulator cells. These cells are initialized to zero and gets incremented by one whenever different transformed line intersects in that cell. Therefore, different points that lie on a straight line in (x, y) plane will form a “cluster of crossings” i.e., gets intersected at respective (ρ, θ) . These clusters are called “Hough peaks” and therefore the lines in (x, y) that corresponds to those Hough peaks are called “Hough Lines”. Only a certain Hough peaks and Hough lines based on a given Threshold value are used. The Hough lines and the Hough peaks that have a value greater than the Threshold level are plotted. The figures (a), (b) shows the Hough line plots. The line segments possess a red and yellow marker to represent the end points of a particular Hough line and the longest line segment is marked with cyan and the remaining with green.

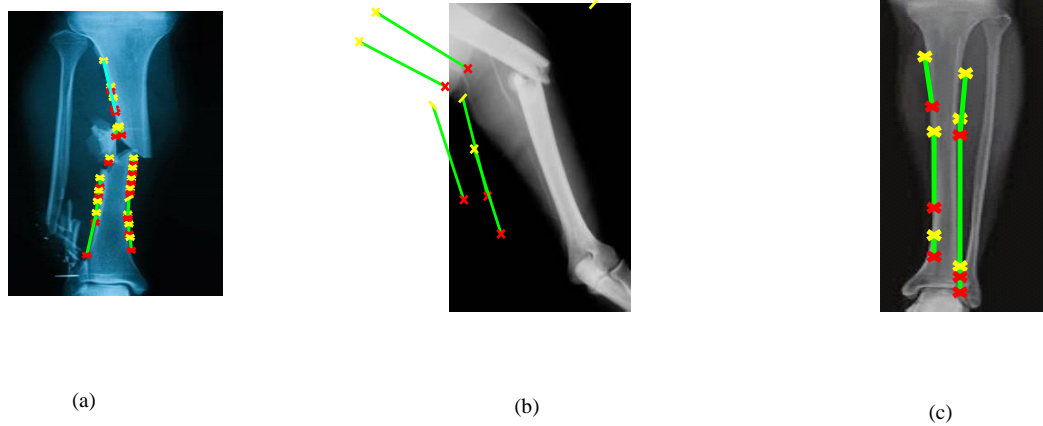


Figure 5. Hough Lines in Images

IV. EXPERIMENTAL RESULTS

Break point detection is performed after plotting the Hough lines for a chosen threshold value. Now, Hough peaks are plotted with respect to θ . Whenever there is more than one major angle contribution it shows more than one Hough peak i.e., two or more break points and only a fractured image will have more than one angle contribution. Thus, we can find whether the fracture is present in the image or not. In the Fig(a) and (b) we have a fracture present in it. Hence, we can find more than one break point whereas in the Fig (c) there is no fracture and thus only one breakpoint. Next, the objective is to plot the area where the fracture has occurred. To do this, firstly, we find the lines corresponding to these breakpoints and convolute this break lines with edge images. Finally, the difference between these convoluted images gives the fractured area. A Break point tolerance of 0.25 can be accepted. The Figures (d), (e) and (f) gives the desired results.

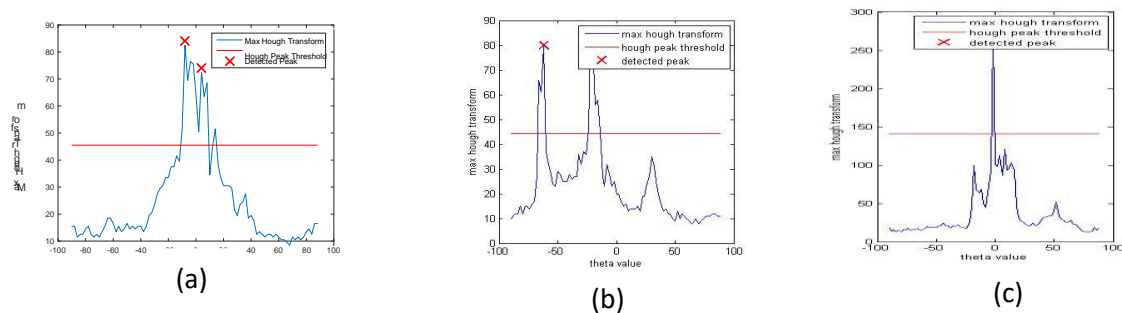


Figure 6. Results of Hough Detection for Fracture and No fracture image



Figure 6. Results for Fracture Detection

V. CONCLUSIONS

This paper presented the image processing technique to detect the bone fracture. The fully automatic detection of fractures in leg bone is an important but it is a difficult problem. This method is not applicable for blur images. According to the test results, the system has been done to detect the bone fracture. A conclusion can be made that the performance of the detection method affected by the quality of the image. The better the image quality, the better the result system got. In future work, focusing on other works like detecting on smaller bone, ankle fractures, etc. may be considered

REFERENCES

- [1] Vijaykumar, V., Vanathi, P., Kanagasabapathy, P. (2010). Fast and efficient algorithm to remove gaussian noise in digital images. *IAENG International Journal of Computer Science*, 37(1).
- [2] Al-Khaffaf, H., Talib, A. Z., Salam, R. A. (2008). Removing salt-and-pepper noise from binary images of engineering drawings. In: *Pattern Recognition. ICPR. 19th International Conference on*, p. 1–4. IEEE.
- [3] Zain, M. L. M., Elamvazuthi, I., Begam, M. (2009). Enhancement of bone fracture image using filtering techniques. *The International Journal of Video and Image Processing and Network Security*, 9 (10).
- [4] Mahmoud Al-Ayyoub, Iamail Hmeidi, Haya Rababaha, *Detecting Hand Bone Fractures in X-Ray Images*, Jordan University of Science and Technology Irbid, Jordan, Volume 4. No.3, September 2013
- [5] S.K.Mahndran, S.Santhosh BaBoo, *An Enhanced Tibia Fracture Detection Tool Using Image Processing and Classification Fusion Techniques in X-Ray Images*, Sankara College of Science and Commerce, Coimbatore, Tamil Nadu, India, Online ISSN: 0975-4172 & Print ISSN: 0975-4350, Volume 11 Issue 14 Version 1.0 August 2011.
- [6] J Samuel Febriarto Kurniawian, I Ketut Gede Darma Putra, et al, *Bone Fracture Detection Using Opencv*, Department of Information Technology, Udayana University, Indonesia, ISSN: 1992-8645, Vol. 64, No.1, June 2014)
- [7] Ms. Senhal Deshmukh, Ms Shivani Zalte, Mr.Shantanu Vaidia, Mr.Parag Tangade: *Bone Fracture Detection Using Image Processing in Matlab*, Internal Journal of Advent Research in Computer and Electronics, 28 March, (2015)
- [8] Tai Peng Tian, Ying Chen, Wee Kheng Leow, et al, *Computing Neck-Shaft Angle of Femur for X-Ray Fracture Detection*, Dept. of Computer Science, National University of Singapore
- [9] Donnelley, M., "Computer aided long-bone segmentation and fracture detection", Ph.D. thesis, Flinders University of South Australia, 2008.
- [10] Shubhangi D.C, Raghavendra S.Chinchansoor, P.S Hiremath, *Edge Detection of Femur Bones in X-ray images – A Comparative study of Edge Detectors*, Department of Computer Science, Poojya Doddappa Appa College of Engineering, Gulbarga – 585103 India, Volume 42-No.2, March 2012
- [11] Raman Maini, Himanshu Aggarwal, March 2009 *Study and Comparison of Various Image Edge Detection Techniques* International Journal of Image Processing (IJIP).
- [12] Ekşi, Z., Çakiroğlu, M. "Performance evaluation of the popular segmentation algorithms for bone fracture detection", 2. World Conference on Information Technology, Antalya, Turkey, 23-27 November 2011..
- [13] S. E. Lim, Y. Xing, Y. Chen, W. K. Leow, T. S. Howe, and M. A. Png, "Detection of femur and radius fractures in x-ray images," in *Proc. 2nd Int. Conf. on Advances in Medical Signal and Info. Proc.*, 2004
- [14] Chen, Y., Yap, W.H., Leow, W.K., Howe, T.S. and Png, M.A. (2004) Detecting femur fractures by texture analysis of trabeculae, *Proc. Int. Conf. on Pattern Recognition*
- [15] Lim, V.L.F., Leow, L.W., Chen, T., Howe, T.S. and Png, M.A. (2005) Combining classifiers for bone fracture detection in X-ray images, *IEEE International Conference on Image Processing, ICIP 2005*, Vol. 1, Pp.I - 1149-1152

- [16] Nathanael .E. Jacob, M. v. Wyawahare: Survey of Bone Fracture Detection Techniques, International Journal of Computer Applications, Volume 71-No.17, June, (2013).
- [17] Tian, T. (2002). Detection of femur fractures in x-ray images. Master's thesis, National University of Singapore, Singapore.
- [18] Rafael C. Gonzalez, Richard E. Woods Digital Image Processing, Second Edition, Prentice Hall, 2002.