X-Ray (2D) and CT-Scanned (3D) image matching for person identification

Supervisor: Professor Md. Haider Ali
Co-Supervisor: Dr. Jia Uddin

Khondker Fariha Hossain 13301103
Israt Fariha Emi 13301080
Moumita Liza 13301105

Department of Computer Science and Engineering,
BRAC University

Submitted on: 21st August 2017

DECLARATION

We declare that this thesis is based on the work results done by ourselves. Other's Materials and data which we used for research are mentioned by reference. This thesis, neither in whole nor in part, has been previously submitted for any degree.

Signature of Supervisor

____

Professor Md. Haider Ali

Signature of Co-Supervisor

____

Dr. Jia Uddin

Signature of Authors

____

Khondker Fariha Hossain

____

Israt Fariha Emi

____

Moumita Liza

ACKNOWLEDGEMENT

We would first like to express our sincere gratitude to our supervisor Professor Md. Haider Ali and co-supervisor Dr. Jia Uddin for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped us in all the time of research and writing of this thesis.

Besides my advisors, our sincere thanks goes to Samiul Islam, Lecturer of the department of computer science and engineering, BRAC University for his continuous support and help throughout the thesis.

We would also like to thank the Dhaka Medical College Hospital authority for providing us with the necessary data of X-ray and CT-scan image. Without these data we would not be able to complete our thesis.

Finally, we must express our very profound respect to my parents and friends for providing us with unfailing support and continuous encouragement

TABLE OF CONTENTS

CHAPTER 4

CHAPTER 5

CHAPTER 6

LIST OF FIGURE

LIST OF TABLE

Abstract

Development in medical science has created a vast research area in medical image processing. Superimposition of different type of diagnostic images like X-ray, CT scan, MRI scan which provide accuracy in identification of body organ has brought a great help in medical surgery or surgical planning. Our aim is to improve accuracy in human identification and space and time optimization for X-ray and CT scanned images which we use as our dataset. We have applied different edge detection method to find out the best boundary image and also have superimposed X-ray and CT scanned edge images of same patient to improve the level of accuracy. Then, thresholding is run on X-ray and CT scan images using various way to compare the value obtained in different method. In addition, sum of the Euclidean distances has been measured which gives almost similar value for different patients' CT-scan and X-ray image. On the other hand, it produces quite different value applying on same patient's dataset. Moreover, we optimize time along with space by compressing 2D matrix into a single column matrix implementing training method.

Chapter 1

Introduction

Medical imaging technology has brought a great revolution in medical science. Images, taken through X-ray, CT scan etc. help doctors to detect the diseases accurately. These images are also very important to get the exact view of the organ of human body which is needed to be operated. [1]Moreover, medical images help to examine an area where patient experiencing pain or discomfort and diagnose, monitor, and treat many medical conditions [2] In addition, CT scan image shows bone, soft tissue and blood vessels all at the same time on a single picture. To provide a perfect and proper image of organ we are going to superimpose the X-ray image on the CT scan image.

X-ray unit produce rays which are passing through the body and release a picture. Those pictures are put into a CD and then converted into a digital image. This X-ray digital image file format is DICOM which is known as digital imaging and communications in medicine. X-ray produces 2D image using attenuation of the X-ray beam. [3]Ct-Scan are special X-ray tests that produce cross-sectional images of the body using X-rays and a computer.

Ct examines help doctors to visualize little tumors which cannot be seen from a plain view of the X-Ray beam.[5] Apart from standard single beam X-rays, CT-scan release a series of limited beams to generate greater amount of details. CT scanners can differentiate tissues inside a rigid organ. CT-scan data is transmitted to a computer, which builds up a 3D cross-sectional picture of the part of the body and displays it on the screen. The file format of CT-scan is 3D picture of the organ which is in a DICOM file format. CT imaging provides very good anatomic information [RR-3378]

X-ray image is two dimensional which is a great challenge for superimposing on CT scan image as CT scan images are three dimensional. It is also very difficult to handle with lots of data of another method to superimpose of the X-ray image on CT scan image. After completing the superimposition process we get a new image of the body part containing both X-ray and CT scan data together which gives a clear visualization for identifying organs. We run thresholding and Euclidean distance and compare the difference of them. For time and space optimization we applied training method and convert the 2D matrix into single column matrix.

The superimposed image will provide the accurate view from every angle of the body part which needs surgery. Moreover, proper angle, diameter, density of that organ can be find out from this picture. For the surgery of a body organ, every point can be measured from the new image. However, the final superimposed image will be beneficial for the surgeon and the surgical patient also.

Chapter 2

2. Literature Review

For image processing one of the fundamental operation is to detect the edge of the image which reduces the amount of pixel in the image. We have used five edge detection algorithm.

i. Canny Edge Detector

ii. Sobel Operator

iii. Prewitt Operator

iv. Robert Edge Detector

v. Log Filter

2.1 Canny edge detection

For image processing one of the fundamental operation is to detect the edge of the image which reduces the amount of pixel in the image. Canny edge detector is a multistage computational algorithm to detect the edge of an image.[5] This edge detector takes grayscale image as an input and creates a picture demonstrating positions of tracked intensity discontinuities. [6]

Canny edge detection has five sequential stages to for detecting edge of an image.

1. Smoothing is the first step of this detection algorithm. It is usual that image will contain some level of noise which needs to be reduced first. Using Gaussian filter original

grayscale image is smoothed where standard deviation is f σ = 1.4 and 5x5 Gaussian filter is used.[7]

2. Second stage of this algorithm is to calculate the gradient magnitude and direction of the image to find out where there is the most grayscale intensity. High magnitude indicates the edges of the image and the direction shows the orientation of the image. The magnitude of gradient is $m = \sqrt{G_x^2 + G_y^2}$. The direction of gradient is $\theta = arctan(\frac{G_y}{G_x})$. Here, G_x and G_y are the X and Y derivatives at the point being considered. These equations are calculated with the help of Sobel edge detector.[8]

3. On the third stage of the algorithm is called non maximum suppression where the main aim to sharpen the blur edges by deleting everything else except all local maxima. For this purpose, gradient direction θ converted to near 45◦ and then positive and negative gradient directions are compared with current pixel. Edge strength value are preserved if the current pixel is largest otherwise it is removed. [7]

4. In fourth stage, thresholding is done. Here, direction information and lower threshold will be used to build the edge. The process is:

   i. If the current one is not edge, then check next pixel.

   ii. If the current pixel is not an edge, check the next one.

iii. If it is an edge, check the two pixels in the direction of the edge. If one of them or both,

- Have the direction in the same bin as the central pixel
- Gradient magnitude is greater than the lower threshold
- They are the maximum compared to their neighbors (non-maximum suppression for these pixels), then these pixels are marked as edge pixel

Loop will be continued until there are no changes in the image. When the image stops changing that is the edge of the image. [8]

2.2 Sobel Operator

The Sobel operator is a discrete differential operator.[9] This method performs 2D spatial slope estimation on a picture and furthermore it stresses areas of high spatial frequency that compare to edges.[10]. It is named after Irwin Sobel and Gary Feldman, colleagues at the Stanford Artificial Intelligence Laboratory (SAIL). It was co-developed with Gary Feldman at SAIL. Sobel and Feldman presented the idea of an "Isotropic 3x3 Image Gradient Operator" at a talk at SAIL in 1968. [11]. Regularly it is used to locate the approximate absolute gradient magnitude at each point in an input grayscale image. [10] The operator utilizes two 3x3 kernels: one calculate the angle in the x-direction, while the other one calculate the angle in the y-direction. [9] One direction is simply the other rotated by 90º.[10] The masks can be applied independently to the input image, to process differentiate measurements of the gradient component in each orientation that is Gx and Gy.[10] These can be mixed collectively to locate the absolute magnitude of the gradient at each point and the orientation of that gradient. The gradient magnitude is given below in Figure 1 [10].

$$|G| = \sqrt{G_{x^2} + G_{y^2}}$$

-i	0	i
-j	0	j
-i	0	i

G_x

i	j	i
0	0	0
-i	-j	-i

G_y

Although typically, an approximate magnitude is computed using:

$|G| = |Gx| + |Gy$………………………………………………………… (i)

Which is much faster to compute [10].

The angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by:

$\alpha = \arctan(Gy/Gx) - 3\pi/4$………………………………………………... (ii)

In this situation orientation 0 is taken to imply that the path of max contrast from black to white runs from left to proper on the photograph, and other angles are measured anti-clockwise from this.[2] Frequently, this supreme magnitude is the main result the client sees the two segments of the angle are helpfully figured and included a single information picture utilizing the pseudo convolution operator appeared in Figure 2[10].

N_1	N_2	N_3
N_4	N_5	N_6
N_7	N_8	N_9

Utilizing this the approximate magnitude is given by:

$$|G| = |(N1 + 2xN_2 + N_3) - (N_7 + 2xN_8 + N_9)| + (N_3 + 2x\,N_8 + N_9) - (N_1 + 2xN_4 + N_7)| \quad \text{(iii)}$$

Because of Sobel operator smoothing impact (Gaussian smoothing), it is less touchy to noise show in pictures. Then again, smoothing influences the exactness of edge identification. As such, the Sobel technique does not create picture with high exactness for edge discovery, however its quality is satisfactory enough to be utilized as a part of various applications[9].

2.3 Prewitt Operator

Prewitt is a discrete separation operator, processing an estimation of the gradient of the picture intensity function. The Prewitt operator is based totally on convolving the picture with a small, separable, and integer valued filter in horizontal and vertical instructions and is therefore noticeably less expensive in phrases of computations like Sobel operator. The Prewitt operator was developed by Judith M. S. Prewitt[12]. It is an approximate approach to estimate the value and orientation of the edge. The Prewitt operator uses the same equations as the Sobel operator, apart from the constant k.

Unlike Sobel operator, Prewitt operator does not place any emphasis on pixels which can be closer to the center of the masks. The vertical part is calculated with kernel Gx and the horizontal aspect is calculated with kernel Gy. |Gx| + |Gy| provide an indication of the intensity of the gradient in the cutting-edge pixel [10]. The Gradient magnitude is shown in the below Figure.

-i	0	i
-i	0	i
-i	0	i

G_x

i	i	i
0	0	0
-i	-i	-i

G_y

M↓ x = (a↓ 2 + ca↓ (3) + a↓ (4) (a↓ 0 + ca↓ (7) + a↓ (6) (5) M↓ y = (a↓ 6 + ca↓ (5) + a↓ (4) (a↓ 0) + ca↓ (1) + a↓ (2) (6)…………………………………………………..…………. (iv)

Prewitt edge operator gives better performance than that of Sobel operator [12].

2.4 Robert Edge Detector

Robert technique a picture right into a line drawing, transform the line drawing right into a 3-dimensional representation and finally display the 3-dimensional structure with all the hidden lines removed, from any point of view. The Roberts pass set of rules performs a 2-D spatial gradient convolution at the photo. The principle idea is to deliver out the horizontal and vertical edges individually after which to put them collectively for the resulting edge detection.[13] It plays an easy, brief to compute, 2-D spatial gradient dimension on a photo. Pixel values at every point in the output constitute the predicted absolute value of the spatial gradient of the input image at that factor. The operator consists of a pair of 2×2 convolution kernels as shown in figure. One kernel is virtually the other circled via 90°, that is very much like the Sobel operator.[2] The gradient magnitude of the Roberts operator Is shown in the following figure[10].

i	0
0	-i

G_x

0	i
-i	0

G_y

These kernels are designed to reply maximally to edges strolling at forty five° to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be carried out separately to the enter picture, to produce separate measurements of the gradient factor in every orientation (call these Gx and Gy).[2] The Gx picture will enunciate diagonals that run from the top-left to the bottom-right where as the Gy picture will deliver out edges that run top-right to bottom-left.[13] These can then be blended together to locate the absolute significance of the gradient at each factor and the orientation of that gradient [14]. The gradient importance is given through:

$$|G| = \sqrt{Gx^2 + Gy^2} \quad \text{(v)}$$

Although typically, an approximate magnitude is computed using

$$|G| = |Gx| + |Gy| \quad \text{(vi)}$$

which is much faster to compute. [14]

Often, the absolute value is the best output the user sees the two components of the gradient are without difficulty computed and introduced in a single skip over the input photo, the use of pseudo convolution operator proven in figure 5 [10].

N_1	N_2
N_3	N_4

Using this mask the approximate magnitude is given by:

$$|G| = |N_1 - N_4| + |N_2 - N_3|$$
$$\frac{\partial y}{\partial x} = y(i, j) - y(i+1, j+1)$$
$$\frac{\partial y}{\partial x} = y(i+1, j) - y(i, j+1)$$
………………………………………………………………. (vii)

The angle of orientation of the edge giving rise to the spatial gradient (relative to the pixel grid orientation) is given by [2]:

$$\theta = \arctan(Gy / Gx) - 3\pi / 4$$
………………………………………………………………… (viii)

2.5 LOG Filter

In LOG filtering image needs to be smoothed first by removing noise and then computes the second derivatives. Edges are distinguished as the locus of focuses where the second subordinate of the filtered picture is equivalent to zero. Here, the input is a grayscale image and output will be also a grayscale image [15].

The Laplacian *L(x,y)* of an image with pixel intensity values *I(x,y)* is given by:

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \quad \text{(ix)}$$

From the set of discrete pixels, discrete convolution kernel need to be found out which is approximate to the second derivatives. Two commonly used discrete approximations to the Laplacian filter which are noise sensitive given below [16].

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

For smoothing the image Gaussian filtering is used before implementing Laplacian filter which will minimize the noise level [17]. Gaussian smoothing filter can be combined with the Laplacian filter and then this hybrid filter is convolved with the image to achieve the required result [15].

LOG function centered on zero and with Gaussian standard deviation as the form:

$$LoG(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \text{(x)}$$

After edge detection thresholding is another important procedure in image processing. For thresholding we have used following methods.

2.6 Mean

Mean is most fundamental of all statistical degree. Means are frequently used in geometry and evaluation; a huge range of manner have been developed for those purposes. In contest of image processing filtering the usage of imply is classified as spatial filtering and used for noise reduction. In this section we've got mentioned approximately diverse sort of mean and analyzed their use for disposing of various kind of noise in image processing [18].

2.6.1 Arithmetic Mean

The arithmetic mean filter, additionally referred to as averaging filter out, operates on an sliding 'm×n' window by means of calculating the average of all pixel values in the window and changing the middle pixel value in the destination photo with the end result. Its mathematical components is given as follows [18].

$$f(x,y)=\frac{1}{mn}\sum_{(r,c)\in W} g(r,c) \qquad \text{(xi)}$$

Where 'g' is the noisy image, f(x,y) is the restored image, and 'r' and 'c' are the row and column coordinates respectively, within a window 'W' of size 'm×n' where the operation takes place.

The mathematics imply clear out reasons a certain quantity of blurring (proportional to the window length) to the image, thereby decreasing the results of noise and local variations. It can be used to reduce noise of different kinds, but works exceptional for Gaussian, uniform, or Erlang noise [18].

2.6.2 Geometric Mean

The geometric mean filter out is a variation of the mathematics mean filter and is more often than not used on images with Gaussian noise. This filter is known to hold picture detail better than the arithmetic imply clear out. Its mathematical system is as follows [18].

$$f(x,y) = \left[\prod_{(r,c)\in W} g(r,c) \right]^{1/mn} \qquad \text{(xii)}$$

In this case each restored pixel is given by the product of the pixel in the sub image window raised to the power '1/mn'.

2.6.3 Harmonic Mean

The harmonic mean filter out is some other variation of the arithmetic suggest filter out and is useful for snapshots with Gaussian or salt noise. Black pixels (pepper noise) are not filtered. The filter's mathematical system is as follows [18].

$$f(x,y) = \frac{mn}{\sum_{(r,c)\in W} \left(1/g(r,c) \right)} \qquad \text{(xiii)}$$

2.6. Contraharmonic Mean

The contra-harmonic mean filter is another variant of the arithmetic mean filter and is normally used for filtering salt or pepper noise (but now not each). Snap shots with salt noise can be filtered the usage of terrible values of R, whereas those with pepper noise can be filtered the use of wonderful values of R. The filter's mathematical formulation is [18],

$$f(x,y)=\frac{\sum_{(r,c)\in W}\left(g(r,c)\right)^{R+1}}{\sum_{(r,c)\in W}\left(g(r,c)\right)^{R}} \qquad \text{...} \qquad (xiv)$$

Where R is the order of filter.

2.7 STD (Standard Deviation)

It's a maximum widely used degree of variability or diversity utilized in statistics. In terms of photograph processing it shows how plenty variant or "dispersion" exists from the common (suggest, or anticipated cost). A low general deviation suggests that the facts factors have a tendency to be very close to the suggest, while excessive trendy deviation shows that the information points are spread out over a massive range of values. Mathematically widespread deviation is given by [18]

$$\tilde{f}(x,y)=\sqrt{\frac{1}{mn-1}\sum_{(r,c)\in W}\left(g(r,c)-\frac{1}{mn-1}\sum_{(r,c)\in W}g(r,c)\right)^{2}} \qquad \text{...} \qquad (xv)$$

A well-known deviation filter calculates the usual deviation and assigns this value to the center pixel inside the output map. Because it has functionality in measuring the range, it is able to be utilized in part sharpening, as depth degree get modifications at the brink of photograph via huge value [18].

Standard deviation filters may be beneficial for radar snap shots. The interpretation of radar snap shots is regularly difficult: you cannot rely upon spectral values due to again scatter (go back of the pulse sent by way of the radar). This often causes a variety of 'noise'. By using a fashionable deviation filter out, you'll be capable of apprehend a few styles [18].

2.8 OTSU Method

Otsu method is one of the most hit techniques for image thresholding due to its simple calculation [19]. Otsu method was proposed by Scholar Otsu in 1979. It is a sort of global thresholding in which it relies upon best on grey value of the picture [20]. Otsu calculates a global threshold with the aid of accepting the existence of two classes, foreground and background pixels, and pick the threshold that minimizes the interclass variance of the edge black and white pixels [3].

Algorithm:

The Otsu's binarization algorithm consists of the following steps [21]:

1. Read a grayscale image.

2. Calculate image histogram.

3. Select a threshold and referred as t, then Calculate foreground variance and then calculate background variance.

4. Calculate Within-Class variance.

5. Repeat steps 3 and 4 for all possible threshold value.

6. Final global threshold, T = threshold in MIN(Within-class variance)

7. Binarize Image = gray scale image > T

Formulation:

Considering, the pixels of a given picture be represented in L gray levels [1, 2,…,L]. The number of pixels at level i is denoted by ni and the total number of pixels by N = n1 + n2 + + nL. In order to simplify the discussion, the gray-level histogram is normalized and regarded as a probability distribution [20] :

$$pi = ni / N, \quad pi > 0 \sum_{i=1}^{L}, Pi = 1 \qquad \text{(xvi)}$$

We divide the pixels into two classes CO and C1 (background and objects, or vice versa) by a threshold at level k; CO denotes pixels with levels [1, , k], and C1 denotes pixels with levels [k + 1,…. , L]. Then the probabilities of class occurrence and the class mean levels, respectively, are given by

$$\omega_0 = Pr\,(Co) = \sum_{i=1}^{k} pi = \omega(k)$$

$$\omega_1 = Pr\,(C1) = \sum_{i=K+1}^{L} pi = 1 - \omega(k)$$

and

$$\mu_0 = \sum_{i=1}^{k} i \Pr(\, i \mid C_0\,) = \frac{\mu(k)}{\omega(k)}$$

$$\mu_1 = \sum_{i=k+1}^{L} i \Pr(i \mid C_1) = \mu_{T-}\, \mu(k)/\, 1 - \omega(k)$$

……………..……….. (xvii)

Where,

$$\omega(k) = \sum_{i=1}^{k} p_i$$

………………………………………………………………………… (xviii)

And,

$$\mu(k) = \sum_{i=1}^{k} i p_i$$

………………………………………………………………………. (xix)

Which are the zero th and the first-order increasing moments of the histogram up to k th level, and

$$\mu_T = \mu(L) = \sum_{i=1}^{L} i p_i \qquad \text{...} \qquad (xx)$$

This is the total mean level of the original picture. We can verify for any value of k :

$$\omega_0\mu_0 + \omega_1\mu_1 = \mu_T \,, \quad \omega_0 + \omega_1 = 1$$

The class variance is given by,

$$\sigma_0^2 = \sum_{i=1}^{k}(i-\mu_0)^2 \Pr(i\,|C_0) = \sum_{i=1}^{k}(i-\mu_0)^2\, p_i/\omega_0$$

$$\sigma_1^2 = \sum_{i=k+1}^{L}(i-\mu_1)^2 \Pr(i\,|C_1) = \sum_{i=k+1}^{L}(i-\mu_1)^2\, p_i/\omega_1 \qquad \text{..................} \qquad (xxi)$$

These required second order cumulative moments. To measure the class separability at threshold level k,

$$\lambda = \sigma_B^2 / \sigma_W^2 \,, \; k = \sigma_T^2 / \sigma_W^2 \,, \; \eta = \sigma_B^2 / \sigma_T^2 \qquad \text{......................................} \qquad (xxii)$$

Where,

$$\sigma_W^2 = \omega_0\, \sigma_0^2 + \omega_1 \sigma_1^2$$

$$\sigma_B^2 = \omega_0\,(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2$$

$$= \omega_0\, \omega_1\,(\mu_1 - \mu_0)^2 \qquad \text{.................................} \qquad (xxiii)$$

And,

$$\sigma_T^2 = \sum_{i=1}^{L}(i-\mu_T)^2\; P_i \qquad \text{...} \qquad (xxiv)$$

are the within class variance, the between-class variance, and the total variance of levels, respectively. well threshold classes would be separated in gray levels, and this threshold is the best threshold.

$$\sigma_W^2 + \sigma_B^2 = \sigma_T^2 \quad \text{...} \quad \text{(xxv)}$$

It shows $\sigma T2$ is independent of k. But the function of $\sigma w2$ and $\sigma B2$ .It also shows that $\sigma w2$ is based on the second-order statistics (class variances), while $\sigma B2$ is based on the first-order statistics (class means).

The optimal threshold k* that maximizes η or equivalently maximizes $\sigma B2$ is selected in the following sequential search by using the simple cumulative quantities.

$$\eta(k) = \sigma_B^2(k) / \sigma_T^2$$

$$\sigma_B^2(k) = \frac{[\mu_T\, \omega(k) - \mu(k)]^2}{\omega(k)[1-\omega(k)]} \quad \text{...} \quad \text{(xxvi)}$$

And the optimal threshold k* is,

$$\sigma_B^2(k*) = max_{1 \le k < L}\, \sigma_B^2(k) \quad \text{...} \quad \text{(xxv)}$$

From the problem, the range of k over which the maximum is sought can be restricted to,

$$S* = \{k;\ \omega_0\, \omega_1 = \omega(k)[1 - \omega(k)] > 0,\ or\ 0 < \omega(k) < 1\} \quad \text{................} \quad \text{(xxvi)}$$

We remember it as a powerful range of the gray-stage histogram, usually recall the maximum threshold price.

2.9 Euclidian Distance Measuring

In image processing esteeming distance between two points is important. The obtained values can be utilized for deciding whether two points are close to each other or not [22]. The Euclidean distance is the distance between two points in Euclidean space [23].

Basically, in Euclidean n-space, if the two points are x= (x_1, x_2…) and y= (y_1, y_2….) then distance would be [24],

$$d(x,y) := \sqrt{(x_1-y_1)^2+(x_2-y_2)^2+\cdots+(x_n-y_n)^2} = \sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \quad \text{…………..} \qquad \text{(xxvii)}$$

For 2D the Euclidean distance between two points a = (ax, ay) and b = (bx, by) is [23]

D(a,b)=√{(bx-ax)^2+(by-ay)^2}…………………………………………………….. (xxviii)

To calculate the Euclidean metrics following steps are required:

Assuming, a set of data point is X = {x1,x2,x3,……..,xn} , and V = {v1,v2,…….,vc} is the set of centers.

i. Select cluster centers ‘c’ randomly.

ii. Then, using the Euclidean distance metric determine the distance between each data point and cluster centers [25]

$$Dxy = \sqrt{\sum_{k=1}^{m} (Xik - Xjk)^2} \quad \text{……………………………………………………….} \qquad \text{(xxix)}$$

iii. Data point is assigned to the cluster center whose distance from the cluster center is minimum of all the cluster centers.

iv. If ‘Ci’ is the number of data points in i th cluster, new cluster center is determined with [4]

$$Vi= \left(\frac{1}{Ci}\right) \sum_{1}^{Ci} Xi \quad \text{(xxx)}$$

v. The distance between each data point and new cluster centers is calculated again [25].

vi. If no data point was reassigned then stop, otherwise repeat steps from iii to v [25].

2.10 TRAINING METHOD

After superimposition of X-ray and CT-Scan image, 2D grayscale image is generated. This grayscale image needs to be binarized where we get matrix of binary value correspondent to pixel value of the grayscale image. This 2D matrix has multiple row and column which is space consuming. For space optimization, we have used training method. At first, binary values are converted into decimal and summed up which generate another 2D matrix including one row and one column. By generating single row and column matrix from multiple row and column we have reduced the space needed for the values.

Chapter 3

3. System Implementation

We use some dataset of 2D image of X-ray and 3D image of CT-Scan of different patient's. We worked a specific angle of an image like front view, side view etc. In here we choose to work with the front view of the person's X-ray and 3D CT-Scan Image. The Images we used are in DICOM format. In next, a number of various edge detection algorithms are run on both the 2D and 3D images. Edge detection algorithms like Canny, Robert, Prewitt, Log are used to get the most accurate and efficient edge detected image. The steps and methods are discussed below on details.

3.1 Work Flow Diagram

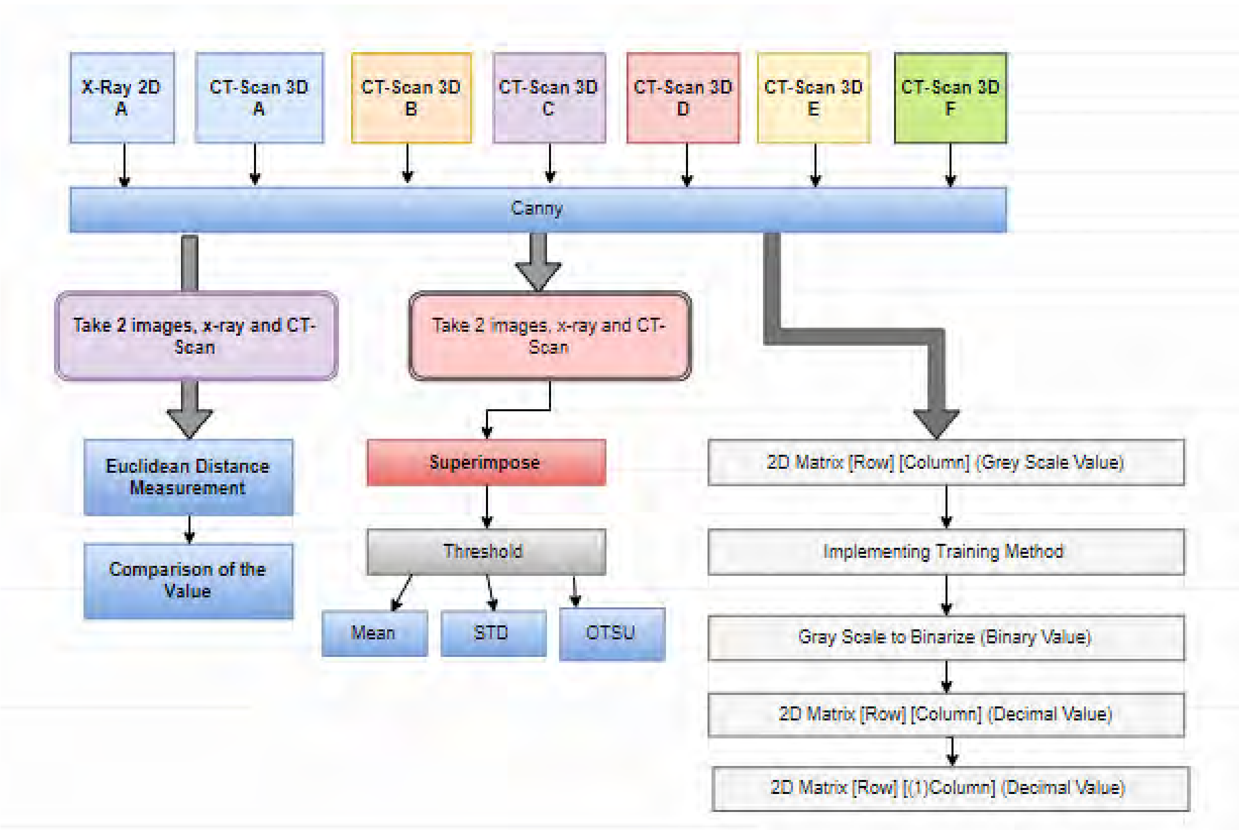

Figure 1: Work flow of System Implementation

3.2 Pseudo Code

1. Start
2. Input image1=X-ray.dicom
3. Input image2=CT-Scan.dicom
4. Convert Grayscale (image1, image2);
5. Edge detestation Canny (image1, image2)

Run Robert, Prewitt, Log, Canny, Sobel and choose Canny for its better output.

6. Super_impose_image= Superimpose (image1, image2)
7. STD_value= STD (Super_impose_image)
8. Mean_value= Mean (Super_impose_image)
9. Otsu_value= Otsu (Super_impose_image)

To find the threshold value,In first two edge detected images are superimposed and Standard Deviation ,Mean and Otsu valured are measured .

10. For the Comaparison of Threshold values step 5,6,7,8 and 9 are repeated for different inputs.
11. Eucledian Distance (image1, image2)

Eucledian Distance are measured from the edge detected images.

12. Input matrix [row][column] (binary value)= binary image(image1, image2)
13. Convert new_matrix [1][column] (decimal value) = input matrix[row][column]
14. End

3.3 Implementation Details

3.3.1View Selection

We get a front of view of 2D X-ray image, which is by default. But in the case of 3D CT-Scan image we get a 3D model view, where we can see the every side of the picture in x, y and z direction. But for superimpose we have to get only one view of the 3D image. For that we extract the 3D image to get all the view of the picture and then we compare every side of the picture with the front view of 2D image. After comparing we get the front view of 3D CT-Scan image which is similar to 2D X-Ray image. These images will be used in the following algorithms and methods.

3.3.2 Edge detection

In this phase we are using five algorithms, which are Sobel, Prewitt, Robert, Canny and LOG to determine edge of the images in a gray scale value and compare them to see which one is better than all other. After comparing all the algorithm one by one we saw that all are giving different gray scale image among them Sobel, Prewitt and Robert are almost but Prewitt is better than Sobel. But one of the most clear gray image with edges we found through Canny algorithm, So, in this paper we choose Canny images to work further on other methods.

3.3.3 Thresholding

In these process first of all we named the pictures according to the person's number. For example we set 2D x-ray image as A for first person, it same for all the process. For 3D Ct-Scan image we set A for first person, set B for second person and set C for third person. Now we superimpose image A of X-ray with image A of CT-Scan. This method will then use for image A and B, and image A and C. After superimposing, we run the thresholding which are Mean finding, STD and OTSU, They are giving different value for each imposing.

3.3.4 Euclidean Distance

Another method we used in our implementation process is Euclidean distance, which produce a distant value of pixels. For this process first we run the method on canny gray scale image for image A of both X-ray and CT-Scan. Then we run the same method again for image A of X-ray and image B of CT-Scan, and also for image A of X-ray and image B of CT-Scan. Basically in all the process we are using same persons X-ray image but with different persons CT-Scan image.

3.3.5 Training Method:

This is the last we try to use Training method. We used canny gray scale image and converted it to a binary image. A image is a 2D matrix for a single row we can get a corresponding decimal value. So we converted the whole 2D matrix[n][n] in a single column matrix[n][1] ,This array looked single column matrix will include the decimal value for the corresponding row of binary value.

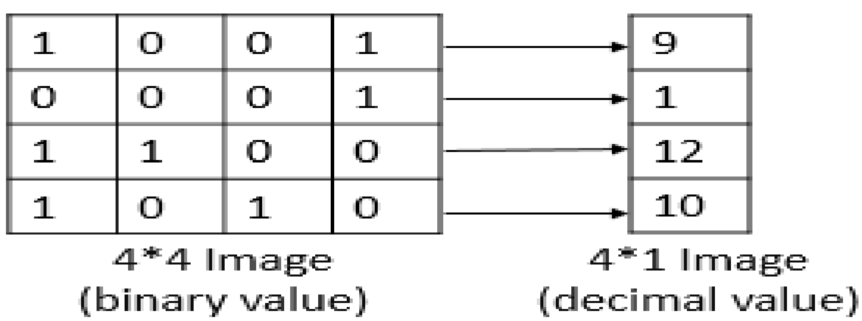

Figure 2: Conversion of the 2D matrix's binary valued row to decimal valued row.

Chapter 4

4. Experimental Result

4.1 Main Dataset

Figure 3:X-Ray Image

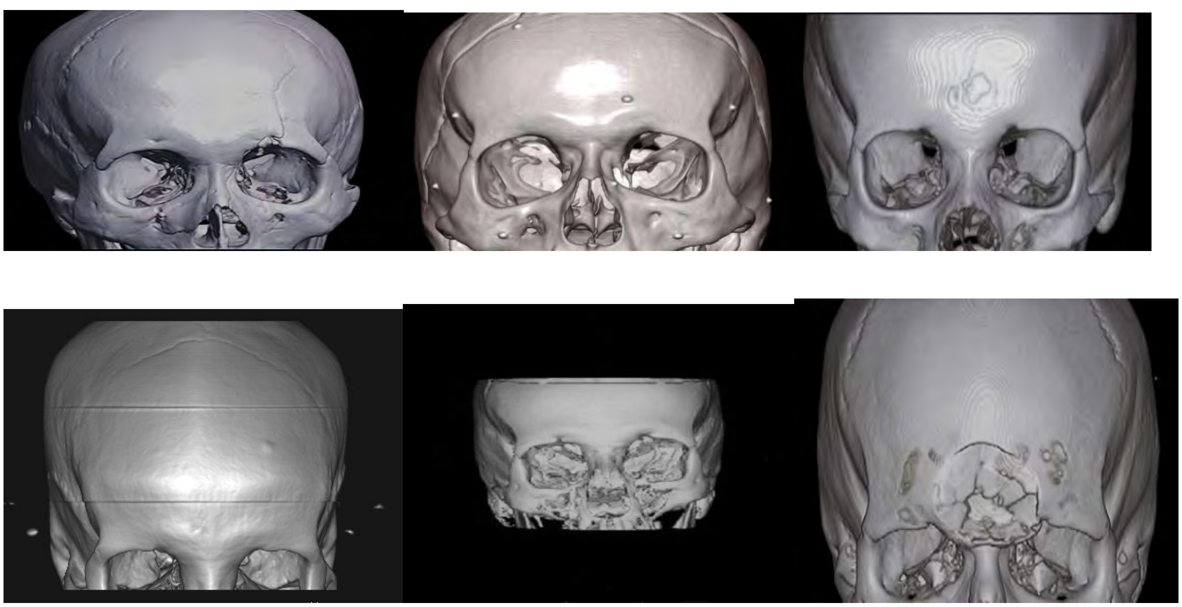

Figure 4: CT-Scan Images

4.2 Implemented Edge detection methods:

Figure 5: Implementation of Edge Detection methods a) Canny b) Log c) Prewitt d) Robert e) Sobel

4.3 Threshold Finding

Superimposed Images

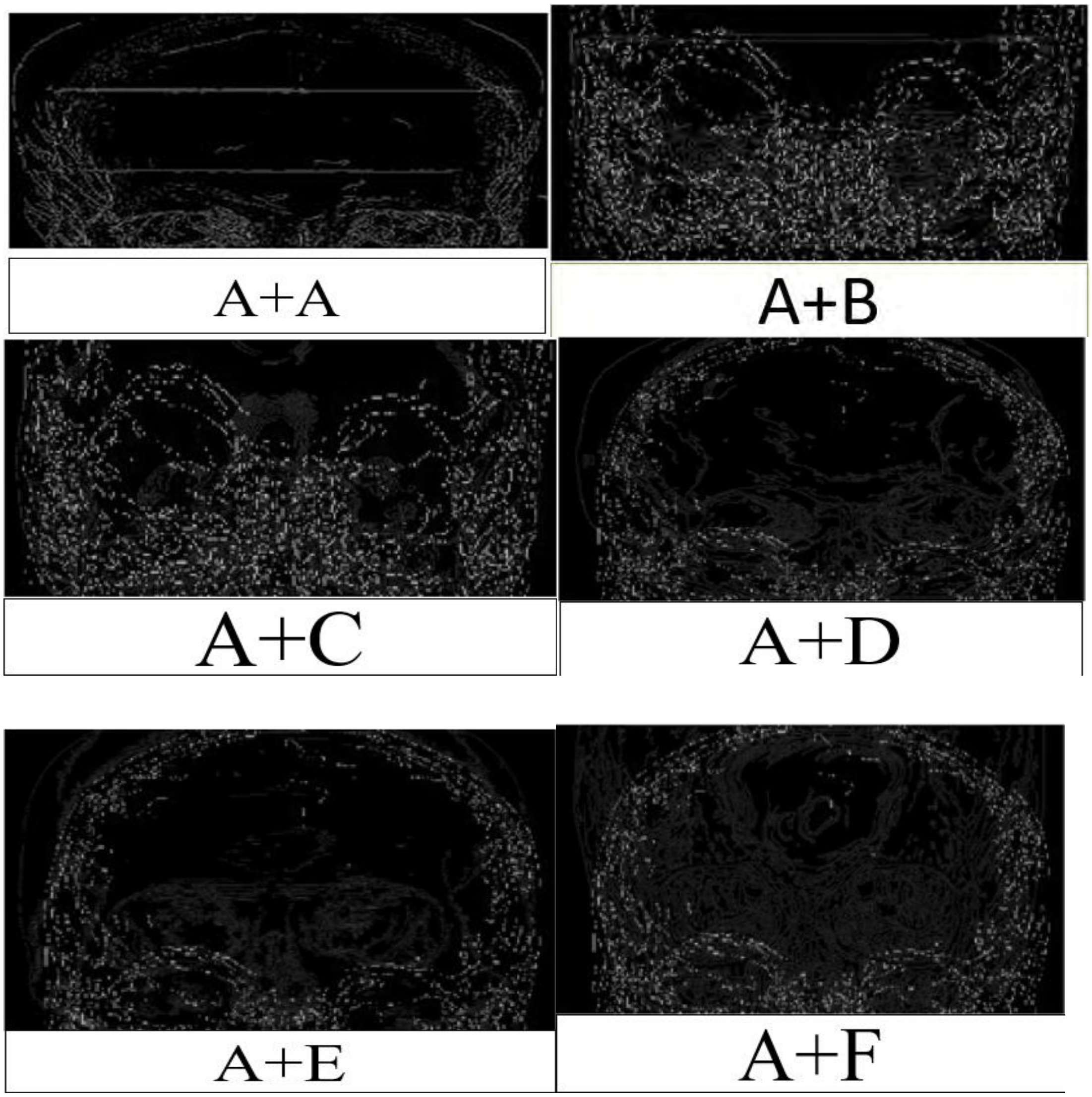

Figure 6: Superimposed Images. Here A=Person 1(X-Ray),

Where B, C, D, E & F are 5 different persons.

4.3 Comparison and Analysis of STD, Mean and Otsu

4.3 (i) Comparison of STD, Mean and Otsu Value

Table 1: Comparison of the values of Mean, STD and Otsu.

(CT+X-Ray)	MEAN	STD	OTSU
A+A (Same person)	8.0124	25.2743	0.3145
A+B	17.7398	40.2895	0.2078
A+C	16.3121	37.0505	0.1804
A+D	12.1045	28.7217	0.1647
A+E	10.3839	24.1580	0.1569
A+F	17.58	29.91	0.1529

4.3 (ii) Analysis of STD, Mean and Otsu Value

- Here we can on the Table () the mean value is 8.0124.Onthe other hand for different person the values are between in the range of [11~18].
- Again in STD for the same person the value is 25.2743 where for different person range is in between [24~41]. In case of STD the difference is not that much variable.
- In Otsu, same persons value is .3143 where others are ranged from [.15~.20]. So we can say that there is a huge difference on the value of same person to others.

4.3 (iii) Comparison of STD, Mean and Otsu Value (Graph View)

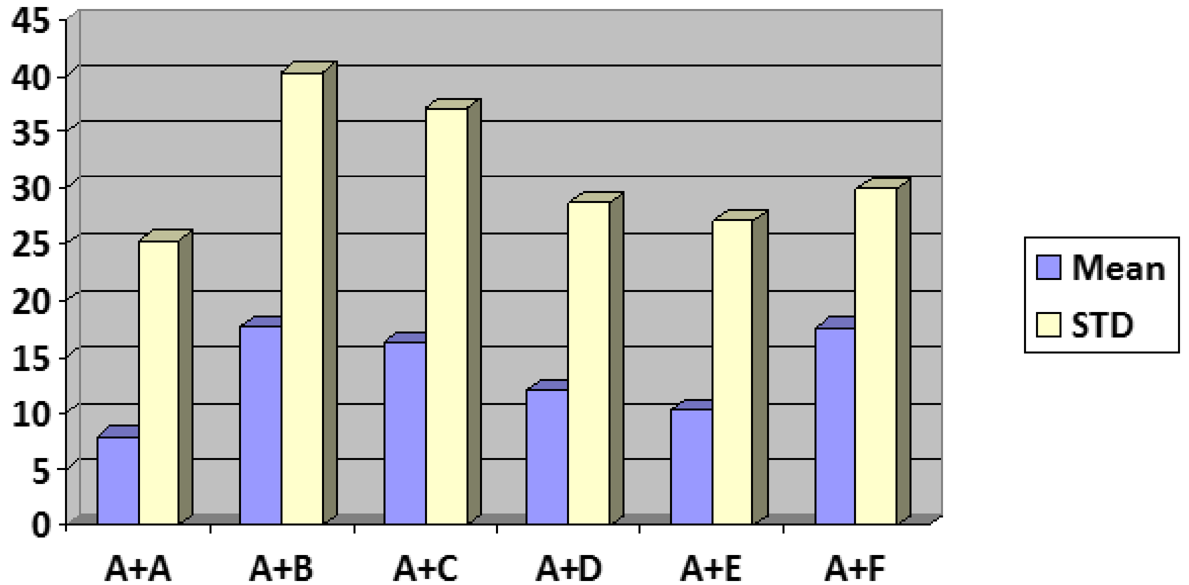

Figure 7: STD value and Mean Value Comparison

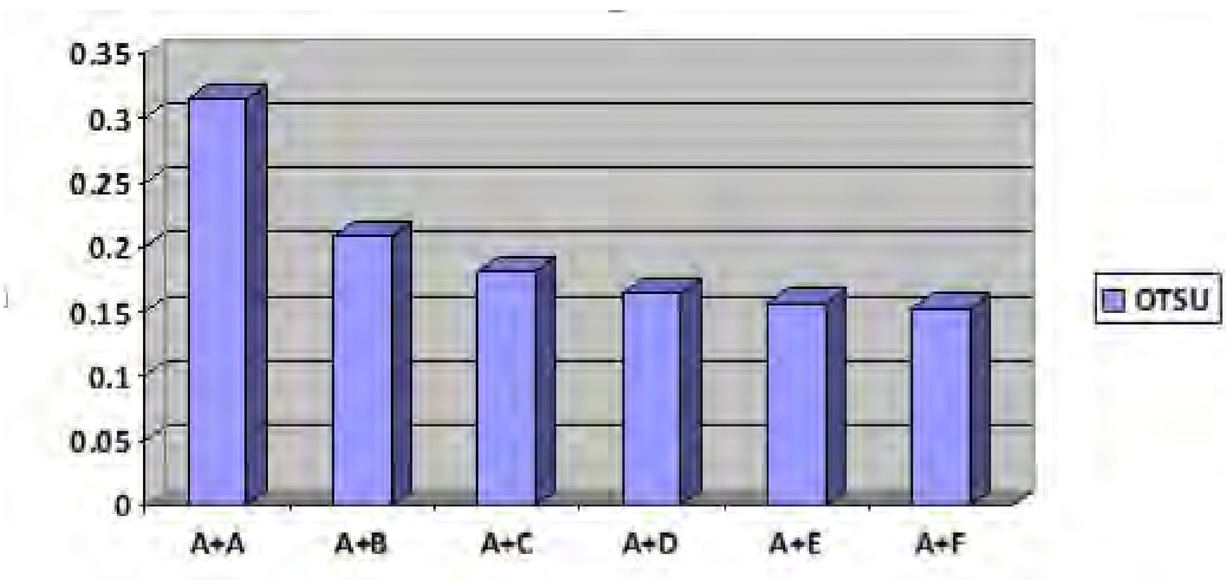

Figure 8: Otsu Value Comparison

4.4 Euclidian Distance Measurement

4.4 (i) Value Comparison

CT+X-RAY	**Euclidian Distance**
A+A(SAME PERSON)	**1.1982e+05**
A+B	**1.1486e+04**
A+C	**1.1403e+04**
A+D	**1.5514e+04**
A+E	**1.3062e+04**
A+F	**1.4575e+04**

Table 2: Comparison of the values of Euclidian Distance

4.4 (ii) Value Comparison (Graph View)

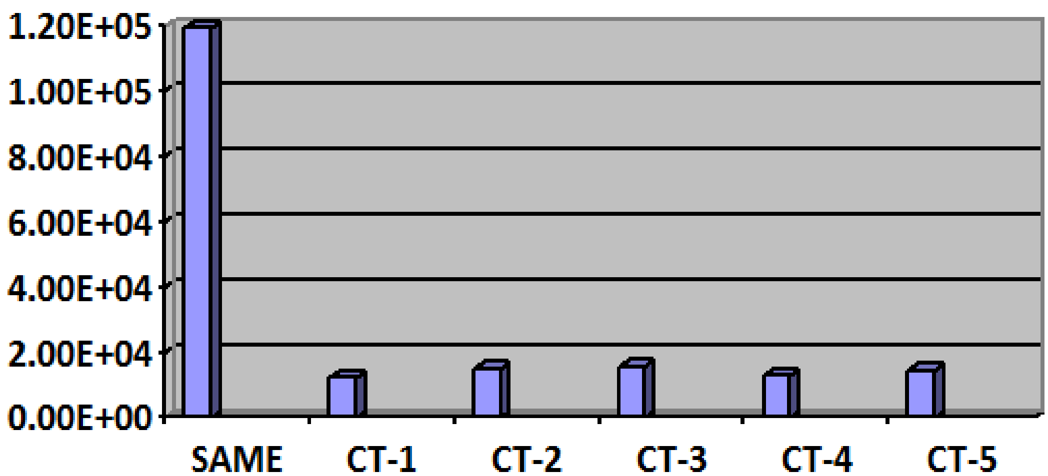

Figure 9: Euclidian Values

4.4 (iii) Analysis of Euclidian Distance Values

From the Table () we can see that for the same person (A+A) the value is really high which is 1.1982e+05 .

Where in other case this Euclidian Distance value differed from range [1.14e+04~1.55e+04]

By comparing these values we can easily identify the person.

Chapter 5

5.1 Conclusion

In this paper we have showed different algorithms and methods for simultaneously computing optimal values for thresholding gradient parameters for different patient's 2D X-Ray image and 3D CT-Scan image. For each example presented here the given methods has successfully determined the number of regions in the image and also detected visually satisfactory threshold for different interfaces. Also the two and three dimensional superimposition method presented here allows the assessment of important structural displacement following surgery and its short and long term stability. Moreover, the Euclidean distance method that extract the exact same value for two and three dimensional image of a same patient's gives us the different value when it comes to different which helps the medical staffs to store the data accurately in their database without any confusion or mistake. The range of this system implementation is not restricted only to the thresholding of gray level picture but it may also cover other cases with the help of more methods, as we wish to implement training method in this case which can be efficient in the matter of time and space for the future medical work. Despite all training, expertise, technical support, and time required, these methodologies seem to have great validity for clinical, scientific and educational orthodontic and surgical application. In the future we will try to make our system more efficient. As we only tested few algorithms and methods due to shortage of dataset and time, in the future we will try to measure the accuracy for most of the edge detection algorithm and threshold method, which maybe come out as a one of the most helpful technique in the future medical imaging technology.

5.2 Future Work

In future a system can be built that will take the raw images of CT-Scan an X-ray. This system will automatically identify the person. So the output will be the identified person. Furthermore, it will save these image data of the person as a 1D matrix. Decrease the time consumption and space optimality will be a major target for the system.

Chapter 6

5. Reference

1. "Why Would Someone Need a CT Scan? | Two Views." Accessed December 13, 2016. http://www.two-views.com/ct/why-need.html#sthash.Dc8fV1JB.6xcxl4Jp.dpbs
2. "X-Ray." Healthline. Accessed December 13, 2016. http://www.healthline.com/health/x-ray.
3. What Is a CT Scan (CAT Scan)? Definition, Contrast, CT vs. MRI. (n.d.). Retrieved December 13, 2016, from http://www.emedicinehealth.com/ct_scan/article_em.html
4. Abdominal CT Scan. (n.d.). Retrieved December 13, 2016, from http://www.healthline.com/health/abdominal-ct-scan
5. John Canny. A computational approach to edge detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on, PAMI-8(6):679–698, Nov. 1986.
6. Homepages.inf.ed.ac.uk. (2017). *Feature Detectors - Canny Edge Detector*. [online] Available at: https://homepages.inf.ed.ac.uk/rbf/HIPR2/canny.htm [Accessed 5 Aug. 2017].
7. Canny Edge Detection. (2009). [ebook] Available at: http://www.yildiz.edu.tr/~mercimek/courses/courses_Spring_2017/kom3212/files/canny.pdf [Accessed 5 Aug. 2017].

8. U. Sinha, "The Canny Edge Detector: Introduction the edge detector - AI Shack - Tutorials for OpenCV, computer vision, deep learning, image processing, neural networks and artificial intelligence.", *Aishack.in*, 2017. [Online]. Available: http://aishack.in/tutorials/canny-edge-detector/. [Accessed: 05- Aug- 2017].
9. D. Kim, "www.egr.msu.edu," [Online]. Available: http://www.egr.msu.edu/classes/ece480/capstone/fall13/group04/docs/danapp.pdf. [Accessed 2017 August 2017].
10. J. Preetha, S. Selvarajan and P. Suresh, COMPARATIVE ANALYSIS OF VARIOUS IMAGE EDGE DETECTION TECHNIQUES FOR TWO DIMENSIONAL CT SCAN NECK DISC IMAGE. IJCSC, 2012.
11. "en.wikipedia.org/wiki/Sobel_operator," [Online]. Available: https://en.wikipedia.org/wiki/Sobel_operator. [Accessed 7 August 2017].
12. "en.wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Prewitt_operator. [Accessed 6 August 2017].
13. O.R. Vincent and O. Folorunso, *A Descriptive Algorithm for Sobel Image Edge Detection*. Proceedings of Informing Science & IT Education Conference (InSITE), 2009.
14. R. Maini and H. Aggarwal, *Study and Comparison of Various Image Edge Detection Techniques*. International Journal of Image Processing (IJIP).
15. P. Sharma, G. Singh and A. Kaur, *Different Techniques Of Edge Detection In Digital Image Processing*. / International Journal of Engineering Research and Applications (IJERA), 2003.

16. "Spatial Filters - Laplacian/Laplacian of Gaussian", *Homepages.inf.ed.ac.uk*, 2017. [Online]. Available: https://homepages.inf.ed.ac.uk/rbf/HIPR2/log.htm. [Accessed: 05-Aug- 2017].
17. S. Liqin, S. Dinggang and Q. Feihu, *Edge Detection on Real Time Using LOG Filter*. Hong Kong: 1994 International Symposium on Speech, Image Processing and N e u r a l N e t w o r k s, 1994.
18. V. Kumar and P. Gupta, *Importance of Statistical Measures in Digital Image Processing*. International Journal of Emerging Technology and Advanced Engineering, vol. 2, no. 8, 2012.
19. P. A. B. Miss Hetal J. Vala, "A Review on Otsu Image Segmentation," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET),* vol. 2, no. 2, 2012.
20. J. Amruta B. Patil, "OTSU Thresholding Method for Flower Image Segmentation," *International Journal of Computational Engineering Research (IJCER),* vol. 6, no. 5, 2016.
21. A. Kumawat R. Bhargava Dr. N. Bhargava, "Threshold and binarization for document image," *Threshold and binarization for document image,* vol. 17, no. 5, 2014.
22. D. Xu and H. Li, *Euclidean Distance Transform of Digital Images in Arbitrary Dimensions*.
23. http://shodhganga.inflibnet.ac.in/bitstream/10603/33597/12/12_chapter4.pdf. [Accessed 7august 2017].
24. "Computer Vision," [Online]. Available: http://computervision.wikia.com/wiki/Euclidean_distance. [Accessed 7 august 2017]
25. A. Singh, A. Yadav and A. Rana, *K-means with Three different Distance Metrics*. International Journal of Computer Applications (0975 – 8887), 2013.