QUANTITATIVE ANALYSIS AND FRACTURE DETECTION OF PELVIC BONE X- RAY IMAGES

Mr.R.Vijayakumar ME
Asst.Proff, Dept of ECE
Mahendra Engineering College
Namakkal , Tamilnadu
e-mail: rvijay.nethra@gmail.com

Mr.G.Gireesh
2nd Year ME COS
Mahendra Engineering College
Namakkal, Tamilnadu
Email:gireeshgopal@gmail.com

***Abstract*— Today bone fractures are very common in our country because of road accidents or through other injuries. The X-Ray images are the most common accessibility of peoples during the accidents. But the minute fracture detection in X-Ray image is not possible due to low resolution and quality of the original X-Ray image. The complexity of bone structure and the difference in visual characteristics of fracture by their location. So it is difficult to accurately detect and locate the fractures also determine the severity of the injury. The automatic detection of fractures in X-Ray images is a significant contribution for assisting the physicians in making faster and more accurate patient diagnostic decisions and treatment planning. In this paper, an automatic hierarchical algorithm for detecting bone fracture in X-Ray image is proposed. It uses the Gray level co-occurrence matrix for detecting the fracture. The results are promising, demonstrating that the proposed method is capable of automatically detecting both major and minor fractures accurately, and shows potential for clinical application. Statistical results also indicate the superiority of the proposed methods compared to other techniques. This paper examines the development of such a system, for the detection of long-bone fractures. This project fully employed MATLAB 7.8.0 (.r2009a) as the programming tool for loading image, image processing and user interface development. Results obtained demonstrate the performance of the pelvic bone fracture detection system with some limitations.**

***Keywords*— X-Ray Image, segmentation, Edge detection, GLCM.**

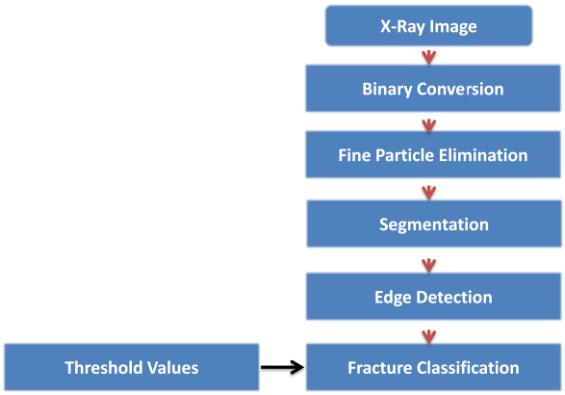
I. INTRODUCTION

Every year, traumatic injuries contribute to cases of death and permanent disability. Patients with bone fractures who go into shock state have a mortality of 30-50%. When combined with other injuries in the body, for example, an abdominal injury, the chance of mortality rises even higher, approaching 100% in some cases. Traumatic injuries in this region can result in severe hemorrhage, multiple organ dysfunction syndromes (MODS), injury to the nerves, internal organ damage, thus resulting in a mortality rate from 8.6% to 50%. Even if injuries at this magnitude don't occur, severe pain and impaired mobility are normally associated with pelvic fractures. Automatic and accurate detection of fractures from segmented bones in traumatic injuries can help physicians detect the severity of injuries in patients. The extraction of fracture features (such as presence and location of the fracture) as well as fracture displacement measurement are all very important for assisting physicians in making faster and more accurate decisions. However, due to low resolution of original images, variations in bone structures, different visual characteristics of fractures by locations and other factors, automatic detection of bone fractures using X-Ray image is not easy to achieve. Fracture detection X-Ray images are still an under-explored field. There are very few direct studies focused on fracture detection in X-Ray images.

II. MATERIALS & METHODS

In this section, we describe the overall system design, image pre-processing techniques, segmentation, results analysis and classification. The input of X-Ray images, in DICOM 3.0 standard format will be interpreted into the developed software. Now we can analyses each step of the system in implementation level and verify the outputs

A. System Design



In order to detect the fracture bone, edge of bone features appears as a vital rule for the classification task. A wide conventional of edge detector have been considered, such as Sobel and Canny techniques have a shortcoming in border energies calculation, as more than two interest plotlines will be mapped within the output image. Thus, we have developed our own unique algorithm using shaft to obtain the edge. Lastly, the resultant region will be undergoing computerized texture analysis using GLCM, where it will be compared with threshold value in order to carry out the classification.

B. Image Pre-processing

Binary Conversion of the image: The image is changed to binary to ease the computing process and maximize the speed

of calculating due to the rapid Boolean operators. By changing to binary, the bone shaft can be separated from the soft tissue shade which can be considered noise during the bone shaft image processing. Suppose threshold value, H each pixel value is compared to H and decision is made to define a new pixel value which is either zero or one of the corresponding pixel in the output binary image. Suppose the output image A and the input image is B and N is the image pixel array. The comparison operator is shown below:

$$A(N) = \begin{cases} 0, if\ B(N) \leq H \\ 1, if\ B(N) \geq H \end{cases} \text{ for } N = E \rightarrow F$$

The threshold value is of extremely significance and need to be carefully selected. The quality of the image depends heavily on this threshold value.

Gray Scale Conversion: In computing, a grayscale or grayscale digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. The gray scale histogram can be used for the further processing of the image like segmentation, edge detection etc. To convert any color to a grayscale representation of its luminance, first one must obtain the values of its red, green, and blue (RGB) primaries in linear intensity encoding, by gamma expansion. For the sRGB color space, the gamma expansion is defined as

$$C_{\text{linear}} = \begin{cases} \frac{C_{\text{srgb}}}{12.92}, & C_{\text{srgb}} \leq 0.04045 \\ \left(\frac{C_{\text{srgb}}+0.055}{1.055}\right)^{2.4}, & C_{\text{srgb}} > 0.04045 \end{cases}$$

Where C_{srgb} is any of the three gamma-compressed sRGB primaries in range [0,1] and $C_{\text{linear}}$is the corresponding linear-intensity value (also in range [0,1]). Then, luminance is calculated as a weighted sum of the three linear-intensity values. For the ITU-R BT.709 primaries, as used in sRGB, the weighting

$$Y = 0.2126\ R + 0.7152\ G + 0.0722\ B$$

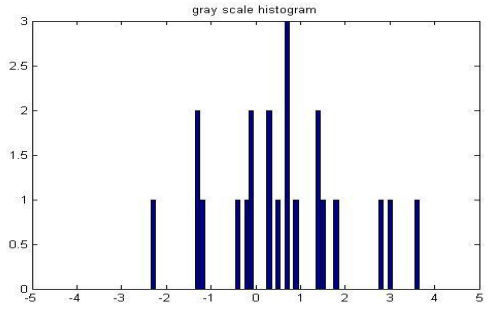


Fig 1 Gray scale histogram image

The reverse is also possible to build a full color image from their separate grayscale channels.

$$Y' = 0.2126R + 0.7152G + 0.0722B$$

Fig 2 Gray scale histogram image

Segmentation of X Ray Image: In computer vision, image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). 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. The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image (see edge detection).

Thresholding: The simplest method of image segmentation is called the thresholding method. This method is based on a clip-level (or a threshold value) to turn a gray-scale image into a binary image. This allows for the removal of other surrounding artifacts which exist in the original image, e.g. X-Ray table, cables, hands, and lower extremities. Next, the image is enhanced to emphasize bone structures. For this technique, image contrast is increased by mapping grayscale values to a wider range; the bone is enhanced to higher intensity.

Region-growing methods: This method takes a set of seeds as input along with the image. The seeds mark each of the objects to be segmented. The regions are iteratively grown by comparing all unallocated neighboring pixels to the regions. The difference between a pixel's intensity value and the region's mean δ is used as a measure of similarity. The pixel with the smallest difference measured this way is allocated to the respective region. This process continues until all pixels are allocated to a region. Seeded region growing requires seeds as additional input. The segmentation results are dependent on the choice of seeds. Noise in the image can cause the seeds to be poorly placed. Unseeded region growing is a modified algorithm that doesn't require explicit seeds. It

starts off with a single region A_1– the pixel chosen here does not significantly influence final segmentation. At each iteration it considers the neighboring pixels in the same way as seeded region growing. It differs from seeded region growing in that if the minimum δ is less than a predefined threshold T then it is added to the respective region A_j. If not, then the pixel is considered significantly different from all current regions A_i and a new region A_{n+1} is created with this pixel.

Edge Detection: Edge detection is a fundamental tool in image processing, particularly in the areas of feature detection and feature extraction, which aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. It can be shown that under rather general assumptions for an image formation model, discontinuities in image brightness are likely to correspond to:

- discontinuities in depth,
- discontinuities in surface orientation,
- changes in material properties and
- Variations in scene illumination.

Edges extracted from non-trivial images are often hampered by fragmentation, meaning that the edge curves are not connected, missing edge segments as well as false edges not corresponding to interesting phenomena in the image – thus complicating the subsequent task of interpreting the image data. There are many methods for edge detection, but most of them can be grouped into two categories, search-based (gradient) and zero-crossing based (Laplacian). Thus, a one-dimensional image f which has exactly one edge placed at $x = 0$ may be modeled as:

$$f(x) = \frac{I_r - I_l}{2}\left(\operatorname{erf}\left(\frac{x}{\sqrt{2}\sigma}\right) + 1\right) + I_l.$$

At the left side of the edge, the intensity is

$$I_l = \lim_{x \to -\infty} f(x),$$

And right of the edge it is

$$I_r = \lim_{x \to \infty} f(x).$$

The scale parameter σ is called the blur scale of the edge.

Sobel operator: The Sobel operator is used in image processing, particularly within edge detection algorithms. Technically, it is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function. The Sobel operator is based on convolving the image with a small, separable, and integer valued filter in horizontal and vertical direction and is therefore relatively inexpensive in terms of computations. This implies that the result of the Sobel operator at an image point which is in a region of constant image intensity is a zero vector and at a point on an edge is a vector which points across the edge, from darker to brighter values.

Mathematically, the operator uses two 3×3 kernels which are convolved with the original image to calculate approximations of the derivatives - one for horizontal changes, and one for vertical. If we define A as the source image, and G_x and G_y are two images which at each point contain the horizontal and vertical derivative approximations, the

$$\mathbf{G}_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * \mathbf{A} \quad \text{and} \quad \mathbf{G}_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * \mathbf{A}$$

computations are as follows:

Where'$*$ 'here denotes the 2-dimensional convolution operation.

For example, $\mathbf{G_x}$ can be written as

$$\begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

The *x*-coordinate is defined here as increasing in the "right"-direction, and the *y*-coordinate is defined as increasing in the "down"-direction. At each point in the image, the resulting gradient approximations can be combined to give the gradient magnitude, using:

$$\mathbf{G} = \sqrt{\mathbf{G}_x{}^2 + \mathbf{G}_y{}^2}$$

The Sobel operator consists of two separable operations. Smoothing perpendicular to the derivative direction with a triangle filter

$$h(-1) = 1, h(0) = 2, h(1) = 1$$

Simple central difference in the derivative direction:

$$h'(-1) = -1, h'(0) = 0, h'(1) = 1$$

Sobel filters for image derivatives in different dimensions with

$x, y, z, t \in (0, -1, 1)$:

1D: $h'_x(x) = h'(x);$

2D: $h'_x(x, y) = h'(x)h(y)$

An example the Sobel kernel :

$$h'_z(:,:,-1) = \begin{bmatrix} +1 & +2 & +1 \\ +2 & +4 & +2 \\ +1 & +2 & +1 \end{bmatrix} \quad h'_z(:,:,0) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad h'_z(:,:,1) = \begin{bmatrix} -1 & -2 & -1 \\ -2 & -4 & -2 \\ -1 & -2 & -1 \end{bmatrix}$$

and the two derivatives G_x and G_y can therefore be computed as

$$G_x = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * \left(\begin{bmatrix} -1 & 0 & 1 \end{bmatrix} * A \right) \quad \text{and} \quad G_y = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} * \left(\begin{bmatrix} 1 & 2 & 1 \end{bmatrix} * A \right)$$

Gray Level Co-occurance Matrix: The Gray Level Coocurrence Matrix (GLCM) method is a way of extracting second order statistical texture features. A GLCM is a matrix where the number of rows and columns is equal to the number of gray levels,G, in the image. The matrix element **P(i, j | Δx,Δy)** is the relative frequency with which two pixels, separated by a pixel distance **(Δx,Δy)**,occur within a given neighborhood, one with intensity and the other with intensity **j**. One may also say that the matrix element **P(i, j| d, θ)** contains the second order statistical probability values for changes between gray levels**i** and**j** at a particular displacement distance **d** and at a particular angle **(θ)**. Given an **M×N** neighborhood of an input image containing G gray levels from **0** to **G−1**, let **f(m, n)** be the intensity at sample **m**, line **n** of the neighborhood.
Then

$$P(i, j \mid \Delta x, \Delta y) = WQ(i, j \mid \Delta x, \Delta y)$$

Where

$$W = \frac{1}{(M - \Delta x)(N - \Delta y)}$$

$$Q(i, j \mid \Delta x, \Delta y) = \sum_{n=1}^{N-\Delta y} \sum_{m=1}^{M-\Delta x} A$$

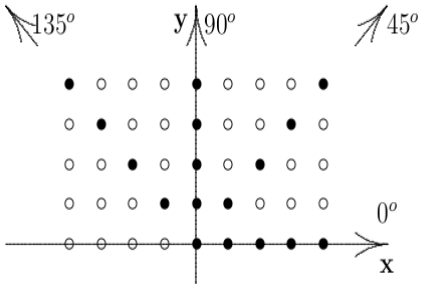
$$A = \begin{cases} 1 & \text{if } f(m,n) = i \text{ and } f(m + \Delta x, n + \Delta y) = j \\ 0 & \text{elsewhere} \end{cases}$$

A small (5×5) sub-image with 4 gray levels and its corresponding gray level coocrurrence matrixP(i, j | Δx= 1,Δy= 0) is illustrated below.

```
IMAGE                  P(i,j;1,0)

0 1 1 2 3              j=0    1    2    3
0 0 2 3 3        i= 0  1/20 2/20 1/20  0
0 1 2 2 3             1    0   1/20 3/20  0
1 2 3 2 2             2    0    0   3/20 5/20
2 2 3 3 2             3    0    0   2/20 2/20
```

Because a G×G matrix (or histogram array) must be accumulated for each sub-image/window and for each separation parameter set **(d, θ),** it is usually computationally necessary to restrict the **(d, θ)**values to be tested to a limited number of values. Figure below illustrates the geometrical relationships of GLCM measurements made for four distances **d(d =max{ | Δx |,| Δy | })** and angles of θ= 0, π/4, π/2 and 3π/4 radians under the assumption of angular symmetry.



A number of texture features may be extracted from the GLCM. **G**is the number of gray levels used; **μ**is the mean value of P. **μx, μy, σx** and **σy** are the means and standard deviations of **Px** and **Py.**

$$P_x(i) = \sum_{j=0}^{G-1} P(i, j)$$

$$P_y(j) = \sum_{i=0}^{G-1} P(i, j)$$

$$\mu_x = \sum_{i=0}^{G-1} i \sum_{j=0}^{G-1} P(i, j) = \sum_{i=0}^{G-1} iP_x(i)$$

$$\mu_y = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} jP(i, j) = \sum_{j=0}^{G-1} jP_y(j)$$

$$\sigma_x^2 = \sum_{i=0}^{G-1} (i - \mu_x)^2 \sum_{j=0}^{G-1} P(i, j) = \sum_{i=0}^{G-1} (P_x(i) - \mu_x(i))^2$$

$$\sigma_y^2 = \sum_{j=0}^{G-1} (j - \mu_y)^2 \sum_{i=0}^{G-1} P(i, j) = \sum_{j=0}^{G-1} (P_y(j) - \mu_y(j))^2$$

$$P_{x+y}(k) = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j) \quad i + j = k$$

- Homogeneity, Angular Second Moment (ASM):

$$ASM = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \{P(i, j)\}^2$$

- Contrast:

$$CONTRAST = \sum_{n=0}^{G-1} n^2 \{ \sum_{i=1}^{G} \sum_{j=1}^{G} P(i,j) \}, \quad |i-j| = n$$

- Local Homogeneity, Inverse Difference Moment (IDM):

$$IDM = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{1}{1+(i-j)^2} P(i,j)$$

- Entropy:

$$ENTROPY = -\sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i,j) \times log(P(i,j))$$

- Correlation:

$$CORRELATION = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{\{i \times j\} \times P(i,j) - \{\mu_x \times \mu_y\}}{\sigma_x \times \sigma_y}$$

- Sum of Squares, Variance:

$$VARIANCE = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (i-\mu)^2 P(i,j)$$

- Sum Average:

$$AVER = \sum_{i=0}^{2G-2} i P_{x+y}(i)$$

III. RESULT AND ANALYSIS

Figure shows part of our experimental results using sample patients‟ data and the obtained findings demonstrated that the state of the art of computerized GLCM method is able to produce accurate border in most of the samples. Figure 3 shows the resultant image conversion into binary form, where the value 0 represents black pixel and value 1 represents the white pixel in the image.

Fig.3 Binary Converted Image

Figure 4 shows the segmented version of the binary converted image. In figure 5 edges are detected from the segmented image.

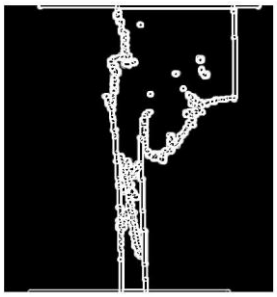


Fig 4 segmented Image

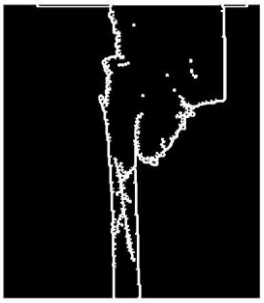


Fig 5 Edge Detected Image

The output of the edge detector consists of a lot of background noise lines. In order to remove these unwanted lines we need to apply a filter at the output of the edge detector. Normally a sobel filter can be applied in cascade with the edge detector for smoothing the output. Figure 6 shows the output of sobel filters.

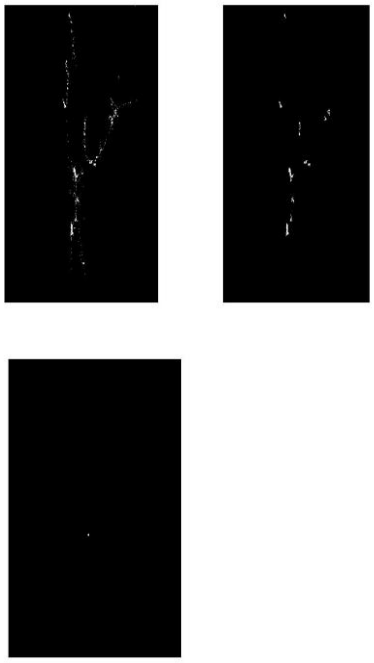


Fig 6 Output of sobel filter 1, 2, 3

Figure 7 shows the fracture detected from the X-Ray image using the algorithm.

Fig 7 extracted the crack.

In GLCM there is a comparison between fractured and unfractured bone by using all the parameters. Figure 8 shows the comparison graph.

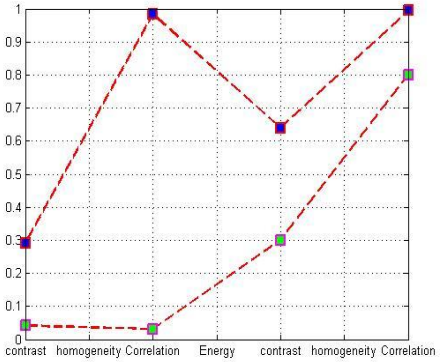


Fig 8 Comparison of different parameters

IV CONCLUSION

We have proposed a method for automated femur bone fracture detection using GLCM computerized techniques. From this method we are able to classify the absence and presence of bone fracture based on the obtained parameter value from GLCM value. The accuracy of the developed algorithm is achieved at least 87 percent which promises an efficient method to recognize bone fracture automatically. Findings show that the system is able to provide consistent and reproducible results.

REFERENCES

[1] Hum Yan Chai, Lai Khin Wee, Tan TianSwee, Sh-HussainSalleh, A.K. Ariff and Kamarulafizam.(2011) 'Gray-Level Co-occurrence Matrix Bone Fracture Detection'

[2] UMD. Ciren report: Consequences and costs of lowerextremity injuries, EMS University of Maryland National Study Center for Trauma, 2005.

[3] M. A. Schiff , A. F. Tencer, and C. D. Mack, "Risk factors for pelvic fractures in lateral impact motor vehicle crashes," Accident Analysis & Prevention, vol. 40, no. 1, pp.387-391, 2008.

[4] S. Ali, T. Pedro, D. Joseph, O. Marcus, I. Kenji, M. R. Daniel, and D. Demetrios. "Predictors of Positive Angiography in Pelvic Fractures: A Prospective Study", Journal of the American College of Surgeons, 207(5): 656- 662, 2008.

[5] M. H. Moghari and P. Abolmaesumi, "Global Registration of Multiple Point Sets: Feasibility and Applications in Multi-fragment Fracture Fixation", Proceedings of MICCAI (2), pp.943-950, 2007.

[6] M. H. Moghari and P. Abolmaesumi, "Global Registration of Multiple Bone Fragments using Statistical Atlas Models: Feasibility Experiments", 30th Annual IEEE International Conference of the Engineering in Medicine and Biology Society, pp.5374 – 5377, 2008.

[7] S. Winkelbach, R. Westphal, and T. Gösling, "Pose Estimation of Cylindrical Fragments for Semi-automatic Bone Fracture Reduction", in Proc. DAGM-Symposium, pp.566-573, 2003.\

[8] Douglas TS, Sanders V, Pitcher R, van As AB. "Early detection of fractures with low-dose digital x-ray images in a pediatric trauma unit", Journal of Trauma Injury, Infection, and Critical Care, 65(1):E4-7, 2008.

[9] M. A. Png, Y. Chen, W. K. Leow,W. Hsu, and T. S. Howe, “Computing neck-shaft angle of femur for x-ray fracture detection,” In Proc. Int. Conf. on Computer Analysis of Images and Patterns (LNCS 2756), pp. 82–89, 2003.

[10]V. L. F. Lum, W. K. Leow, Y. Chen, T. S. Howe, and M. A. Png, “Combining classifiers for bone fracture detection in x-ray images,” IEEE International Conference on Image Processing, vol. 1, pp. I–1149–52, Sept. 2005.

[11] P.K. Sahoo, S. Soltani, and A.K.C.Wong. “A survey of thresholding techniques”. Computer Vision, Graphics andImage Processing, 41:233–260, 1988.

[12] R.M. Haralick and L.G. Shapiro. “Image segmentation techniques”. Computer Vision, Graphics and Image Processing, 29:100–132, 1985.

[13] G.B. Coleman and H.C. Andrews. “Image segmentation by clustering”. Proceedings of IEEE, 67(5):773–785, 1979.

[14] L. I Kuncheva and L. C Jain. “Nearest neighbor
classifier: Simultaneous editing and feature selection”, Pattern Recognition Letters, 20(11–13): 1149–1156, November 1999.

[15] J. Wu, P. Davuluri, K. Ward, C. Cockrell, R. Hobson, K. Najarian, “A New Hierarchical Method for Multi-level Segmentation of Bone in Pelvic CT Scans,” 33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC 11), pp.3399-3402, August 2011.

[16] S. Belongie, J. Malik, and J. Puzicha. “Shape matching and object recognition using shape Contexts,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 4, 2002.

[17] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham “Active shape models - their training and application,” Computer Vision and Image Understanding, vol. 61, no. 1, pp. 38-59, 1995.