Lumbar Spine Image Segmentation using Linked Outlyingness Tree

B. Suresh Kumar

Abstract: ***Image segmentation is the process of partitioning a digital image into multiple segments. The purpose of image segmentation is to partition an image into meaningful regions with respect to a particular application. The image segmentation is used for various applications such as medical images, satellite images, content based image retrieval, machine vision, recognition tasks and video surveillance etc. Many methods such as compression based methods, thresholding, and clustering has been proposed in literature for segmentations. The clustering methods can be divided into two parts, namely supervised and unsupervised. Supervised clustering involves predefining the cluster size for segmenting whereas unsupervised image segmentation segments by its own cluster values. The spine segmentation method validates cluster extraction and subsequently vertebral image is obtained. The previous methods for segmenting images in the medical field are taking more time and less accuracy of vertebral outputs. In order to overcome the disadvantages a new methodologies proposed. In this proposed work three methods have been implemented, namely lumbar spine image Segmentation using linked outlyingness tree. Supervised image segmentation using LOT, Unsupervised. Comparing each other Lumbar spine image segmentation provides the best solution for medical images. The performance results also proved that the proposed system has better performance over other existing algorithms.***

Index Terms: ***Computed Tomography (CT), Magnetic Resonance Image (MRI), Linked Outlyingness Tree (LOT), Robust Outlyingness Ratio (ROR).***

I. INTRODUCTION

Image segmentation is the process of partitioning a digital image into multiple segments. The purpose of image segmentation is to partition an image into meaningful regions with respect to a particular application. The image segmentation is used for various applications such as medical images, satellite images, content based image retrieval, machine vision, recognition tasks and video surveillance. **Lumbar Spine Image Segmentation** is an integrating computer processing of spine MR images will make the diagnosis more accurate, consistent, time and labor saving, and cost reducing. The proposed linked outlyingness tree algorithm includes a model based spinal cord extraction method and the detection of each intervertebral disk along all the vertebrae. It can be useful for quantitative analysis of the spinal disks and vertebrae, image registration with other imaging modalities, such as CT, computer-based iterative spine prescriptions with or without supervision, and image-guided surgery of the spine [1,2].

Manuscript published on 30 November 2017.
*Correspondence Author(s)
B. Suresh Kumar, Assistant Professor, Department of Computer Science, SAN International College of Arts and Science, Coimbatore (Tamil Nadu), India. E-mail: sureshbsk12@gmail.com

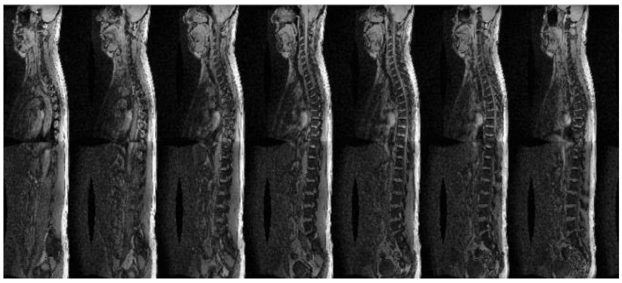

Figure 1: Seven Sections of Sagittal Sequences

In the proposed approach the necessary preprocessing is carried out for the input image to remove noise. After the preprocessing is performed the number of segmentation are decided by the user. Once the number of segment are decided, the LOT method is applied where the ROR value of the image are calculated. If the output does not contain any error the Silhouette analysis used to study the separation distance between the resulting clusters. The silhouette plot displays a measure of how close each point in one cluster is to points in the neighboring clusters[7,8] and thus provides a way to assess parameters like number of clusters visually. This measure has a range of [-1, 1]. Silhouette coefficients (as these values are referred to as) near +1 indicates that the sample is far away from the neighboring clusters. A value of 0 indicates that the sample is on or very close to the decision boundary between two neighboring clusters and negative values indicates that those samples might have been assigned to the wrong cluster. Based on the Silhouette analysis the numbers of segments for the image are obtained.

II. REVIEW OF LITERATURE

In modern medicine, medical imaging has undergone major advancements. Today, this ability to achieve information about the human body has many useful clinical applications. Over the years, different sorts of medical imaging have been developed, each with their own advantages and disadvantages. Certain areas of medicine have made significant headway in adopting digital technology. At the top of this list is radiology—where there has been a serious move toward use of digital modalities such as x-ray, MRI, CT, and ultrasound. X-ray based methods of medical imaging include conventional X-ray, computed tomography (CT) and mammography [3].To enhance the X-ray image, contrast agents can be used for example for angiography examinations. Molecular imaging is used in nuclear medicine and uses a variety of methods to visualize biological processes taking place in the cells of organisms.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Small amounts of radioactive markers, called radiopharmaceuticals, are used for molecular imaging. Other types of medical imaging are magnetic resonance imaging (MRI) and ultrasound imaging. Unlike conventional X-ray, CT and Molecular Imaging, MRI and ultrasound operate without ionizing radiation. MRI uses strong magnetic fields, which produce no known irreversible biological effects in humans.

In this work segmentation via clustering method named Adaptive Fuzzy K-means (AFKM) clustering is used to segment the MRI brain image into three different regions. The AFKM method is proposed to prove that it can classify and segment the (A.P. Zijdenbos and B.M. Dawant, 1994) MRI brain image better than conventional method. AFKM clustering algorithm is combination of KM, MKM and FCM clustering[4,5]. The features of AFKM are to provide a better and more adaptive clustering process.

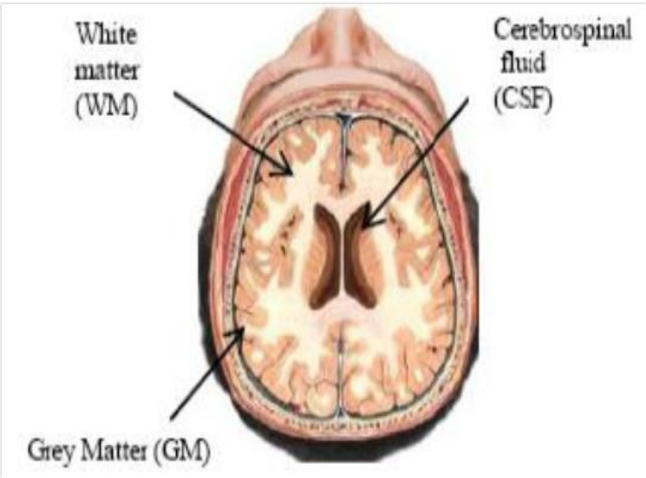

Figure 2: The Normal Brain MRI Image

In medical field, Medical Resonance Image (MRI) is one of the methods [6] used to detect abnormalities in human body. The clustering algorithm for image segmentation was introduced to the MRI images in order to segment the image. In this paper, a new method of clustering algorithm based segmentation known as technique is recommended. The segmentation technique used to be implemented medical image like MRI[2]. It use to exquisite soft tissue contrast between normal tissue and pathologic tissue. The proposed method for this paper is then comparing with conventional method known as Fuzzy C-means (FCM).

Images of brain MRI are obtained from internet database. The images are processed with AFKM and FCM clustering algorithm [9] and comparison is made between the two clustering algorithms. The flow chart for the whole process is depicted in Figure 3.

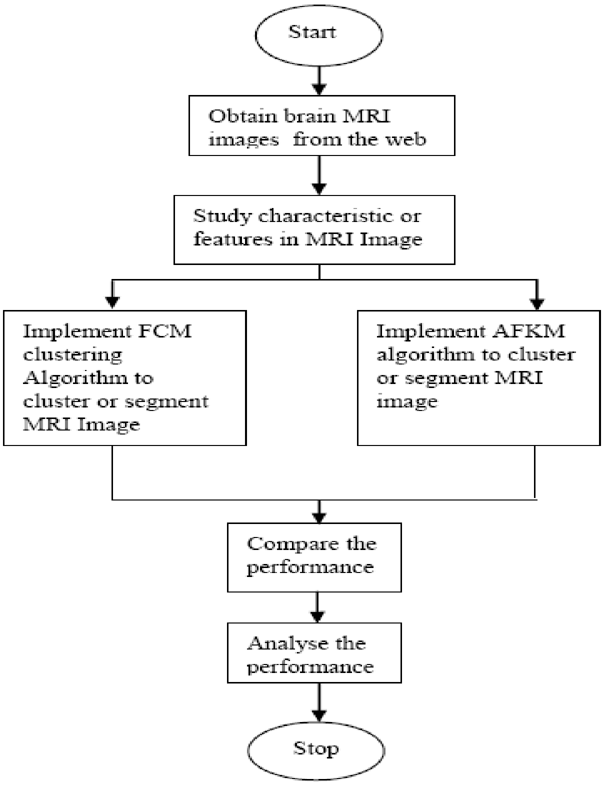

Figure 3: Flow Chart of the Segmentation Clustering Algorithm

III. PROPOSED METHODOLOGY

Enhanced Noise Removal Mechanism for Lumbar Spine Image Preprocessing

A. Noise Models

The image degradation is noise model is discussed as follows

Noise Model 1 Salt and pepper noise with equal noise probability: If [0 255] denote the dynamic range of y', i.e., 0 <= Mij <= 255 for all (i,j), then they are denoted by Salt-and-pepper noise: the gray level of y at pixel location (i j) is illustrated in the equation 1.

$$Y_{ij} = \begin{cases} 0 & \text{with probability } p; \\ M_{ij} & \text{with probability } 1 - p - q; \\ 255 & \text{with probability } q; \end{cases} \quad (1)$$

Where s = p + q denotes the salt-and-pepper noise level.

Noise Model 2 Salt and pepper noise with unequal noise probability White pixels more than black pixels: For the Model 2, it is similar to equal probability noise model, except that each pixel might be corrupted by more number of "salt"(255) noise than " pepper"(0) noise with unequal probabilities. Let P1 and P2 be the probability of occurrence of salt (255) and pepper (0) respectively.

$$Y_{ij} = \begin{cases} P1 & \text{for } X=0; \\ 1-P & \text{for } X= M_{ij}; \\ P2 & \text{for } X=255; \end{cases} \quad (2)$$

Where is the noise density P=P1+P2 and P1≠P2 where P1 >P2.

Noise Model 3 Salt and pepper noise with unequal noise probability black pixels more than white pixels: For the Model 3, it is similar to Model 2,

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

might be corrupted by more number of "Pepper"(0) noise than " salt"(255) noise with unequal probabilities. Let P1 and P2 be the probability of occurrence of salt (255) and pepper (0) respectively.

$$Y_{ij} = \begin{cases} P1 & \text{for } X=0; \\ 1-P & \text{for } X=M_{ij}; \\ P2 & \text{for } X=255; \end{cases} \qquad (3)$$

Where is the noise density P=P1+P2 and P1≠P2 where P2 >P1.

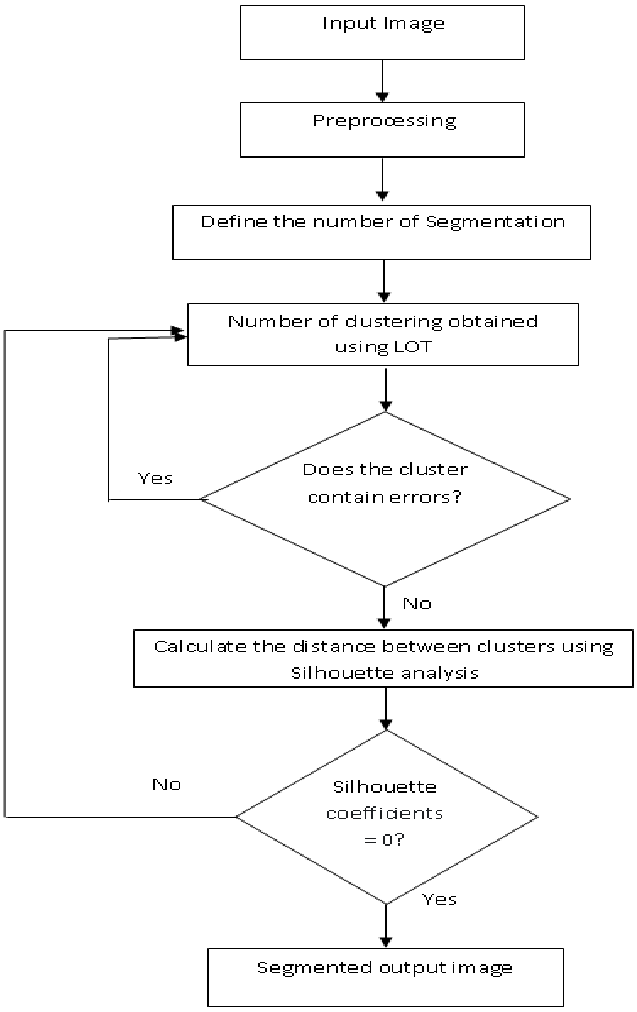

Figure 4: Outline of the Proposed Methodology for Medical Spine Image Segmentation using Linked Outlyingness Tree

B. The Proposed Algorithm for Noise Removal

The proposed algorithm for the removal of salt and pepper noise in images is implemented as follows. If the image is a gray scale image, then the algorithm is applied directly on images. In case of color images, the image is split into corresponding red, blue and green planes. The algorithm is applied on individual planes and later concatenated.

Step 1: Find the mean of non noisy pixels in an image by checking each pixel of the image with 0 or 255 (Which are termed as noisy candidate from the noise model given in Section 2).This mean is referred to Global trimmed Mean.

Step 2: Choose 2-D window of size 3x3. The processed pixel in current window is assumed as Pxy.

Step 3: Convert sorted 2D array into array. Arrange the 1D data in increasing order which is given by S.

Step 4: Check the processed pixel Pxy for 0 or 255

Step 5: If the processed pixel holds 0 or 255 it is considered to be a noisy pixel.

Step 6: Now check the 4 neighbors of the processed pixel for 0 or 255. If all the neighbors are found to be noisy (i.e., holding 0 or 255), replace corrupted pixel with mean of the 4 neighbors

Step 7: If any of the 4 neighbors is not noisy the corrupted pixel is replaced with Unsymmetrical trimmed mean. For higher noise densities there may be a chance that all the pixels of the current processing window might be noisy. Hence it is not possible to calculate the unsymmetrical trimmed mean. So under above stated conditions noisy pixels are replaced with global mean instead of Unsymmetrical trimmed mean.

Step 8: If the Processed pixel does not hold 0 or 255 it is considered as non noisy pixel and hence left Unaltered.

Step 9: Move the window to the next pixel. The above steps from 2 to 8 and is repeated for the entire image.

The processed pixel is checked for low (0) or high (255) values of the gray level values. This process is done on entire pixels in the image. The large matrix refers to image and values enclosed inside a rectangle is considered to be the current processing window. The element encircled refers to processed pixel. The above discussed methodology is illustrated as below. Lease inserts your figures with "inline wrapping" text style, as in this template.

$$\begin{pmatrix} 155 & 255 & 255 & 0 & 255 \\ 0 & 255 & 123 & 255 & 255 \\ 255 & 164 & 255 & 255 & 123 \\ 0 & 122 & 255 & 124 & 255 \\ 0 & 255 & 255 & 255 & 255 \end{pmatrix} \Rightarrow \begin{pmatrix} 155 & 255 & 255 & 0 & 255 \\ 0 & 147 & 123 & 255 & 255 \\ 255 & 164 & 255 & 255 & 123 \\ 0 & 122 & 255 & 124 & 255 \\ 0 & 255 & 255 & 255 & 255 \end{pmatrix}$$

Corrupted image Segment **Restored Image Segment**

Figure 5: Illustration of Case 1

Case 1: Pixel is noisy, some of the four neighbors are noisy, and the processed pixel is 0 which is considered to be noisy. Now check the 4 neighbors of the processed pixel which is given as 0, 123, 164, and 255. Some of the four neighbors are also noisy. Arrange the data inside the current processing window in increasing order.

Unsorted Array: 155 255 255 0 255 123 255 164 255

Sorted Array: 0 123 155 164 255 255 255 255 255

The number of noisy candidate is less than three inside the current processing window. The processed pixel is termed as noisy and the noisy pixel is replaced by unsymmetrical trimmed mean, which is evaluated as follows find the mean of uncorrupted pixel (which is trimmed mean(123, 155, 164) resulting in 147 which replaces the corrupted pixel 0 as illustrated in Figure 4.

Case 2: Pixel is noisy, some of the four neighbors are noisy (But all are noisy), all the elements inside the window are noisy. This case deals with the condition that the processed pixel is noisy (which is 255), now check for the 4 neighbors of the processed pixel (all are 255), Here all the neighbors of the processed pixel are noisy. Count the number of noisy pixels inside the current processing window (which is 9 for this case). We cannot apply unsymmetrical trimmed mean because there is no data uncorrupted to find the trimmed mean. Hence we find global mean of the image.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Consider in the above corrupted image segment (in this case it is assumed to be 5x 5 but in real time it may take the form (512 x512) or (256 x 256)). Global mean is calculated by finding the mean of all the uncorrupted pixels of the given image segment (in case of original simulation this operation is carried out for the entire image) which are indicated in square box. The corrupted pixel is replaced by the global mean, which is given as (123+172+204)/3=166 (123, 172 204 are uncorrupted candidates of the image segment) as shown in Figure 6.

$$\begin{pmatrix} 255 & 255 & 255 & 0 & 255 \\ 255 & 255 & 255 & 255 & 255 \\ 255 & 255 & 255 & 255 & \boxed{123} \\ 0 & \boxed{172} & 255 & \boxed{204} & 255 \\ 0 & 255 & 255 & 255 & 255 \end{pmatrix} \Rightarrow \begin{pmatrix} 255 & 255 & 255 & 0 & 255 \\ 255 & 166 & 255 & 255 & 255 \\ 255 & 255 & 255 & 255 & 123 \\ 0 & 172 & 255 & 204 & 255 \\ 0 & 255 & 255 & 255 & 255 \end{pmatrix}$$

Corrupted image Segment **Restored Image Segment**

Figure 6: Illustration of Case 2

$$\begin{pmatrix} 233 & 0 & 205 & 0 & 255 \\ 225 & 0 & 255 & 255 & 255 \\ 117 & 255 & 165 & 255 & 123 \\ 0 & 122 & 255 & 124 & 255 \\ 0 & 255 & 255 & 255 & 255 \end{pmatrix} \Rightarrow \begin{pmatrix} 233 & 0 & 205 & 0 & 255 \\ 225 & 191 & 255 & 255 & 255 \\ 117 & 255 & 165 & 255 & 123 \\ 0 & 122 & 255 & 124 & 255 \\ 0 & 255 & 255 & 255 & 255 \end{pmatrix}$$

Corrupted image Segment **Restored Image Segment**

Figure 7: Illustration of Case 3

Case 3: Pixel is noisy; all the four neighbors are noisy

In this case the processed pixel is noisy which is (0), now check for the 4 neighbors of the processed pixel (which are 0 255 255 255). It was found that all the 4 neighbors are noisy; Hence find the mean of the 4 neighbors which is illustrated as follows (255+255+255+0)/4= 191. The processed pixel is noisy, the 4 neighbors of the processed pixel is also noisy and hence replace the noisy pixel with mean of the 4 neighbors (which is 191) as shown in Figure 7

Case 4: Processed pixel is not noisy

The processed pixel is 119 which are between 0 and 255. The processed pixel is termed as non noisy and processed pixel is unaltered as shown in figure 5. The algorithm is represented in the form of flowchart as shown in Figure 8.

$$\begin{pmatrix} 0 & 0 & 255 & 0 & 255 \\ 104 & 119 & 255 & 255 & 255 \\ 0 & 103 & 255 & 255 & 123 \\ 0 & 122 & 255 & 124 & 255 \\ 0 & 255 & 255 & 255 & 255 \end{pmatrix} \Rightarrow \begin{pmatrix} 0 & 0 & 255 & 0 & 255 \\ 104 & 119 & 255 & 255 & 255 \\ 0 & 103 & 255 & 255 & 123 \\ 0 & 122 & 255 & 124 & 255 \\ 0 & 255 & 255 & 255 & 255 \end{pmatrix}$$

Corrupted image segment **Restored image segment**

Figure 8: Illustration of Case 4

Efficiency of the original K-means algorithm heavily relies on the initial centroids. In K-means initial cluster centers are generated using random numbers. Due to this the more time is taken for clustering. To improve the performance of the algorithm instead of taking random numbers, the ROR value is considered. In this approach the ROR value of the image are calculated thereby the segments are easily grouped as discussed in the next section.

Algorithm to Calculate ROR Value

Step-1:

The image pixel values represented in a matrix are converted to integers, since the image will be present in uint8 (Unsigned Integer 8 bit) standard which is not convenient for further processing.

A=image pixels in uint8 standard.

A1=double (A) | Now A1 contain integer values of pixels.

Step-2:

The median for the converted value are first calculated.

MED1=Median (A1) | MED1 contains median values of A1.

For example,

A1=1,32,14,15,47,82,24,53,87,69,20;

To find median first we have to sort data in ascending order, Sorted_A1=1,14,15,20,24,32,47,53,69,82,87;

If the total number of elements N, N=odd, then median=middle value of sorted data=even, then median=average of middle two values. MED1= Median (A1) =32. Since N=11.

Step-3:

This median value obtained is again subtracted from the integer value of image and again median is taken for the output.

Sub_A1=absolute value (A1-MED1);

For the above example, Sub_A1= 31,0,18,17,15,50,8,21,55,37,12;

For the obtained new data Sub_A1 again a median value is calculated.

MED2=Median (Sub_A1) For the above example MED2=18

Step-4:

The obtained output is then divided by a value of 0.6457 which is the median of standard normal random variables.

W = MED2 / 0.6457 for the above example it will be, W = 27.87

Finally a matrix of ROR values is obtained by, ROR = (A1-MED)/W

Step-5:

The whole operation is performed for the image values in matrix form. The new output matrix obtained is called ROR value matrix and the values are known as the median absolute deviation or the ROR value.

C. Lumbar Spine Image Segmentation using LOT Method

A strong model for image segmentation is to see the image to segment as a realization of a Markov Random Field Y= $\{Y_s\}_{s\in S}$ defined on the lattice *S*. The random variables $\{Y_s\}_{s\in S}$ have gray level values in the space $\Lambda_{obs}=\{0..255\}$. The configuration set is Ω_{obs}. The segmented image is seen as the realization of another Markov Random Field *X*, defined on the same lattice *S*, taking values in the discrete space $\Lambda=\{1,2,\ldots,K\}$. *K* represents the number of classes or homogeneous regions in the image. To every site $i\in S$ is associated different information: observed information expressed by the random variable Y_i; missed or hidden information, expressed by the random variable X_i. The Random Field *X* is said LOT.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

The segmentation process consists in finding a realization x of X by observing the data of the realization y, representing the image to segment.

We seek a labeling $\hat{x}$, which is an estimate of the true labeling x^*, according to the MAP (Maximum A Posteriori) criterion (maximizing the probability $P(X=x|Y=y)$).

$$\hat{x}=\arg\max_{x\in X}\{P(X=x|Y=y)\} \quad (4)$$

$$P(X=x|Y=y)=\frac{P(Y=y|X=x)P(X=x)}{P(Y=y)} \quad (5)$$

The first term of the numerator describes the probability to observe the image y, knowing the labeling x. Based on the assumption of conditional independence, we get the following formula:

$$P(Y=y|X=x)=\prod_{s\in S}P(Y_s=y_s|X_s=x_s) \quad (6)$$

The second term of the numerator describes the existence of the labeling x. The denominator is constant and independent from x. We have then:

$$P(X=x|Y=y)=KP(Y=y|X=x)P(X=x) \quad (7)$$

$$P(X=x|Y=y)=Ke^{\ln(P(Y=y|X=x))-\frac{U(x)}{T}} \quad (8)$$

$$P(X=x|Y=y)=Ke^{-\Psi(x,y)} \quad (9)$$

The labeling $\hat{x}$ can be found by maximizing the probability $P(X=x|Y=y)$ or equivalently by minimizing the function $\Psi(x|y)$.

$$\Psi(x,y)=-\ln(P(Y=y|X=x))+\frac{U(x)}{T} \quad (10)$$

$$\Psi(x,y)=\Psi_1(y|x)+\Psi_2(x) \quad (11)$$

$$\Psi_1(y|x)=-\sum_{s\in S}\ln(P(Y_s=y_s|X_s=x_s)) \quad (12)$$

$$\Psi_2(x)=\frac{1}{T}\sum_{c\in C}U_c(x) \quad (13)$$

$$\hat{x}=\arg\min_{x\in X}\{\Psi(x,y)\} \quad (14)$$

Assuming that the pixel intensity follows a Gaussian distribution with parameters μ_k (mean) and σ_k^2 (variance). By giving the class label $x_s=k$, we have:

$$P(Y_s=y_s|X_s=k)=\frac{1}{\sqrt{2\pi\sigma_k^2}}e^{-\frac{(y_s-\mu_k)^2}{2\sigma_k^2}} \quad (15)$$

$$\Psi_2(x)=-\frac{\beta}{T}\sum_{s,t\in C_2}\phi(x_s,x_t) \quad (16)$$

The Potts model is often used in image segmentation to privilege large regions in the image. The energy is then:

$$\Psi_2(x)=-\frac{\beta}{T}\sum_{s,t\in C_2}(2\delta(x_s,x_t)-1) \quad (17)$$

$$\Psi(x,y)=\sum_{s\in S}\frac{(y_s-\mu_{x_s})^2}{2\sigma_{x_s}^2}+\ln(\sqrt{2\pi}\sigma_{x_s})-\frac{\beta}{T}\sum_{s,t\in C_2}(1-2\delta(x_s,x_t)) \quad (18)$$

IV. RESULTS AND DISCUSSIONS

1) About the Dataset - Lumbar vertebra segmentation CT image database

The database consists of 12 CT lumbar spine images of 12 subjects, each containing 5 lumbar vertebrae (i.e. levels from L1 to L5). For each vertebra, reference manual segmentation is provided in the form of a binary mask. All images and binary masks are stored in the joint pictures expert group (.jpeg) format.

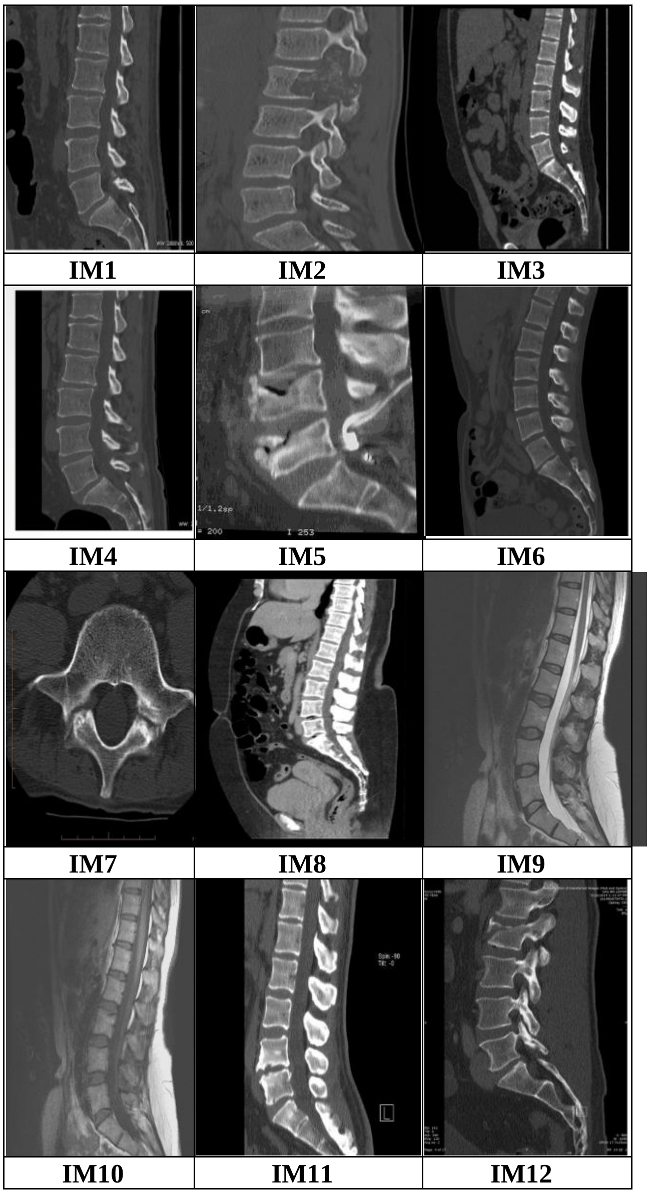

Figure 9: Sample Images

2) Performance Metrics

The following performance metrics are chosen for the comparative analysis of the proposed method with the existing method-True Positive, True Negative, False Positive, False Negative

- Sensitivity
- Specificity
- Accuracy and

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

- Time Taken for Segmentation

Table 1: Performance Analysis of the Proposed LOT Method

S. No	Image	TP	TN	FP	FN	Accuracy	Sensitivity	Specificity
1.	Im1	218704	2790	456	2239	96.945	0.96	0.85
2.	Im2	219019	2654	345	4378	96.972	0.97	0.88
3.	Im3	228886	2856	457	5638	96.716	0.97	0.86
4.	Im4	276334	2658	356	5773	93.962	0.97	0.88
5.	Im5	272581	2856	376	6024	94.847	0.97	0.88
6.	Im6	271445	3896	378	7034	95.283	0.97	0.91
7.	Im7	210063	5635	456	8094	96.103	0.96	0.92
8.	Im8	220019	2746	362	8199	94.763	0.96	0.88
9.	Im9	243648	2895	345	9002	95.839	0.96	0.89
10	Im10	273319	2009	453	9385	96.274	0.96	0.81
11	Im11	206542	2894	365	9991	95.738	0.95	0.88
12	Im12	226208	2745	347	10385	96.382	0.95	0.88

Table 2: Performance Analysis of the Existing method

S. No.	Image	TP	TN	FP	FN	Accuracy	Sensitivity	Specificity
1.	Im1	191153	2851	455	3706	94.372	0.91	0.85
2.	Im2	191255	2817	428	4528	94.839	0.92	0.86
3.	Im3	205717	2886	489	7867	94.103	0.93	0.87
4.	Im4	191073	2866	455	8625	92.178	0.93	0.85
5.	Im5	192906	2878	477	5814	92.874	0.93	0.88
6.	Im6	193486	2853	474	5322	94.026	0.94	0.96
7.	Im7	205575	2834	474	6694	95.014	0.95	0.94
8.	Im8	196036	2883	407	8745	93.563	0.96	0.88
9.	Im9	200220	2884	414	3410	93.902	0.95	0.89
10.	Im10	190975	2803	423	4304	95.003	0.95	0.83
11.	Im11	191153	2851	455	3706	94.184	0.95	0.82
12.	Im12	191255	2817	428	4528	95.037	0.96	0.81

Table 3: Performance Analysis of Time Taken for Segmentation

S. No.	Image	Existing	LOT
1.	Im1	29.336	24.738
2.	Im2	28.989	24.635
3.	Im3	29.025	25.616
4.	Im4	29.566	25.460
5.	Im5	29.387	22.878
6.	Im6	29.702	24.089
7.	Im7	29.342	23.313
8.	Im8	29.478	22.628
9.	Im9	28.894	22.975
10.	Im10	29.433	23.402
11.	Im11	29.336	24.738
12.	Im12	28.989	24.635

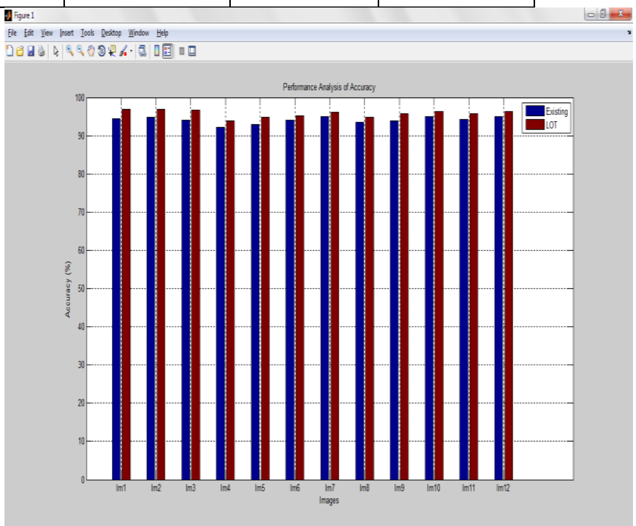

Figure 10: Performance Analysis of Accuracy

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

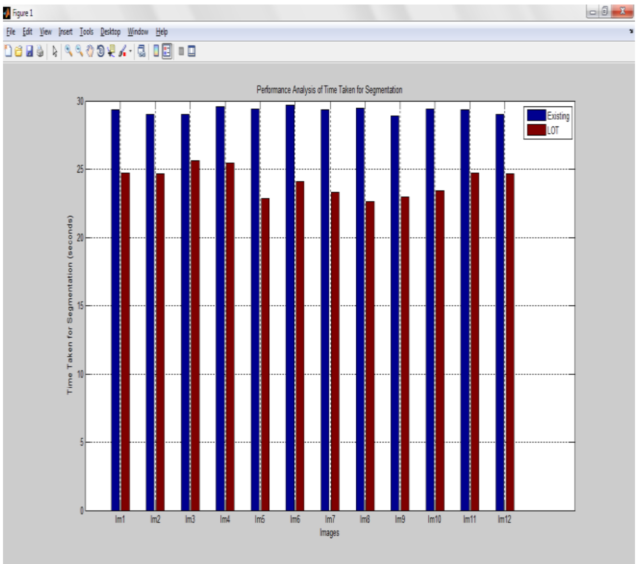

Figure 11: Performance Analysis of Time Taken for Segmentation

V. CONCLUSION

This proposed method lumbar spine image segmentation using linked outlyingness tree shortly coined as LOT. During the first step preprocessing is carried out in order to reduce the noise present in the images. Next an algorithm is proposed for generating ROR value. Then the segmentation has been carried out by using unsupervised semi automatic clustering mechanism. The next chapter proposes unsupervised lumbar spine segmentation using ROR with hidden markow random fields.

REFERENCES

1. A.P. Zijdenbos and B.M. Dawant. Brain segmentation and white matter lesion detection in MR images. Critical Reviews in Biomedical Engineering, 22:401–465, 1994.
2. R. Jack, “Brain and cerebrospinal fluid volume: Measurement with MR imaging,” Radiol., vol. 178, pp. 22–24, 1991.
3. R. Meyer, P. H. Bland, and J. Pipe, “Retrospective correction of intensity inhomogeneities in MRI,” IEEE Trans. Med. Imag., vol. 14, pp. 36–41, Feb. 1995.
4. Pham, “An adaptive fuzzy C-means algorithm for image segmentation in the presence of intensity inhomogeneities,” Pattern Recognition Letters, vol. 20, pp. 57–68, 1999.
5. Pham, “Fuzzy clustering with spatial constraints,” in Proceedings of International Conference on Image Processing (ICPR’02), vol. II, New York, 2002, pp. 65–68.
6. H. Zhang, J. E. Fritts, S. A. Goldman,“Image Segmentation Evaluation: A Survey of unsupervised methods”, computer vision and image understanding, pp. 260-280, 2008.
7. Hui Zhang, Jason E.Fritts,Sally A.Goldman “Image segmentation evaluation: A survey of unsupervised methods”,ELSEVIER, pp:260-280, 2007.
8. J. Noordam, W. van den Broek, and L. Buydens, “Geometrically guided fuzzy C-means clustering for multivariate image segmentation,” in Proceedings of International Conference on Pattern Recognition (ICPR’00), vol. 1, 2000, pp. 462–465.
9. Jain, A.K. and Dubes, R.C. Algorithms for Clustering Data, 1988 (Prentice Hall, Englewood Cliffs, New Jersey, USA).

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)