

Novel mining algorithm for multiple level classification of brain tumors

A. Tamilarasi M

B. Duraiswamy K

Abstract- This paper proposes the various levels of tumors in CT scan brain images, which can assist the medical image diagnosis system. A tumor is a pattern of abnormal cells which are resulting from unwanted changes in the genetic material (i.e. chromosomes) that makes the body cells lose the ability to control their growth. If the tumor does not invade the nearby tissues and body parts, it is called benign tumor, or non-cancerous growth. In contrast, if the tumor invades and destroys the nearby cells, it is called malignant tumor, which is usually life threatening. Brain tumors can be classified into two main categories, primary and secondary. Primary brain tumors (gliomas) are the tumors that start in the brain, whereas the secondary brain tumors result from cancer that starts elsewhere in the body and spreads to the brain (i.e. metastasized). Such a kind of tumors is usually malignant and more common among brain tumor incidences. In Canada, brain cancer that result from malignant tumors causes a large amount of deaths every year; 1650 deaths (out of 2,500 diagnosed cases) were expected in 2005. Regardless of their growth rates, both the malignant and benign tumors have similar effects on the brain. Image segmentation is, arguably, the most important component in the medical image mining process; it is certainly the start point. Segmentation is concerned with the automated division of images into non-overlapping regions. A high speed parallel fuzzy C-means (FCM) algorithm proved to be superior over the other clustering approaches in terms of segmentation efficiency. This proposed algorithm has the advantages of both the sequential FCM and parallel FCM for the clustering process in the segmentation techniques. This algorithm is very fast when the image size is large and it requires less execution time. We have also achieved less processing speed and minimizing the need for accessing secondary storage compared to the previous results. The reduction in the computation time is primarily due to the selection of actual cluster centre and the accessing minimum secondary storage.. Abnormal brain images from four tumor classes namely metastase, meningioma, glioma and astrocytoma are used in this work. In this work, we take advantage of association rule mining. The method proposed here makes use of association rule mining technique to classify the CT scan brain images. It combines the low-level features extracted from images and high level knowledge from specialists. The Experimental results on pre-diagnosed database of brain images shows high accuracy (up to 98%), allowing us to claim that the use of associative classifier is an efficient technique to assist in the diagnosing task. This paper presents a fast association rule mining algorithm which is suitable for medical image data sets. In particular, it assesses the feasibility of using association rule algorithms to extract hidden information from medical image data sets. The main objective of this work is to devise a computational technique that analyzes CT data

by means of association rule mining to come up with a set of information that help in assessing brain tumors.

Keywords- Association rule mining, Computer-aided diagnosis, Associative classifier, FP-Tree, Parallel Fuzzy C means algorithm, Clustering, Decision Tree Classification.

I. INTRODUCTION

A brain tumor is an abnormal growth of cells within the brain or inside the skull, which can be cancerous (malign) or non-cancerous (benign). It is defined as any intracranial tumor created by abnormal and uncontrolled cell division. This type of brain tumor constitutes one of the most frequent causes of death among the human being in the world. Detection of tumor in the earliest stage is the key for its successful treatment. One of the famous method used recently for the screening procedure from the patients include CT-Scan (Computerized Tomography Scan) brain images [1][2]. From the CT-Scan brain images the radiologist will be able to diagnose the abnormalities in the tissues. Even though some 10 to 30% of the tumor cells will not be able to be diagnose correctly. Hence the Computer Aided Diagnosis (CAD) system will assist the physicians as a "Second option" in clearly diagnosing the cancerous cell in CT-Scan brain images [2].

The proposed method classifies the CT-Scan brain images into three types: Normal, benign, and malignant. The normal images depict the cells of healthy patients, benign cells are like cancerous cells but not originally cancerous and third type is malignant cells that depict the original steps for classifying the CT-Scan brain images into the cancerous cells. The proposed method consists of various above mentioned three types of cells they are: preprocessing, feature extraction, segmentation, classification, rule generation and diagnosis.

In this paper image mining concepts have been used. It deals with the implicit knowledge extraction, image data relationship and other patterns which are not explicitly stored in the images. This technique is an extension of data mining to image domain. It is an inter disciplinary field that combines techniques like computer vision, image processing, data mining, machine learning, data base and artificial intelligence [5]. The objective of the mining is to generate all significant patterns without prior knowledge of the patterns [6]. Rule mining has been applied to large image data bases [7]. Mining has been done based on the combined collections of images and it is associated data. The essential component in image mining is the identification of similar objects in

different images. The High speed parallel fuzzy C Means algorithm has been explained for segmentation and removal of inconsistent data from the image [12], [13]. Feature extraction is one of the most important steps in image extraction]. Association rule mining has been used in most of the research for finding the rules for diagnosis in large and small databases. In this proposed method FP-Tree method has been used to find the frequent pattern for building the association rule.

II. SYSTEM DESCRIPTION

The proposed method has been divided into two main phases: the training phase and the test phase. Various techniques followed in these phases are, pre-processing, feature extraction, segmentation, classification, rule generation and Diagnosis. The pre-processing and feature extraction technique are common for both training and test phase.

In the training and test phases the acquired images have been taken for the preprocessing and feature extraction process. The preprocessing has been done by using the median filtering with morphological opening process. Edges are segmented using canny edge detection technique. The regions are extracted in the feature extraction phase. These features are stored in the transactional database. The FP-tree method generates the maximum frequent items that are stored in the transaction database. Association rule can be constructed using maximum frequent itemset that are generated from the FP tree algorithm. The association rules based classifications have been made with the help of decision tree classification. This hybrid approach has been used to classify the CT-Scan brain images into normal, benign and abnormal.

A. Pre-processing

The prime objective of the preprocessing is to improve the image data quality by suppressing undesired distortions (or) enhancing the required image features for further processing. The irrelevant data present in the image has been eliminated using the pre-processing technique. The pre-processing technique eliminates the incomplete, noisy and inconsistent data from the image in the training and test phase. In order to improve the quality of images taken from the CT-scan brain images and to make the feature extraction phase more reliable, pre-processing is necessary. During the digitization process, noise could be introduced that needs to be reduced by applying median filtering techniques. Normalization histogram of the image provides the contrast information and overall intensity distribution. An intensity normalization procedure is carried out by computing an average intensity histogram.

Edge detection is a well-developed field on its own within image processing. Region boundaries and edges are closely related, since there is often a sharp adjustment in intensity at the region boundaries. Edge detection techniques have therefore been used as the base of another segmentation technique. The edges identified by edge detection are often

disconnected. To segment an object from an image however, one needs closed region boundaries.

B. Feature extraction

In pattern recognition and in image processing, feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (much data, but not much information) then the input data will be transformed into a reduced representation set of features (also named features vector). Transforming the input data into the set of features is called *feature extraction*. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

In general, images have the following features – color, texture, shape, edge, shadows, temporal details etc. The features that were most promising were color, texture and edge. *Edge detection* is a fundamental tool in image processing and computer vision, 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.

Low-level features refer to features that can be directly located from the original images. The output of edge detection should be an edge image or edge map, in which the value of each pixel reflects how strong the corresponding pixel in the original image meets the requirements of being an edge pixel. Edges are paid much attention due to its computational cheapness, because differences in pixel values between regions are relatively easy to calculate by considering gradients. More importantly, they provide strong visual clues that can help the recognition process.

An edge may be regarded as a boundary between two dissimilar regions in an image, which may be different surfaces of the object, or perhaps a boundary between light and shadow falling on a single surface. Most edge detection methods work on the assumption that an edge occurs where there is a discontinuity in the intensity (or depth) function or a very steep intensity (or depth) gradient in the image. Edges can be described using three essential parameters, including direction, position, and strength. The **Canny edge detection** operator was developed by John F. Canny in 1986 and uses a multi-stage algorithm to detect a wide range of edges in images.

Canny's aim was to discover the optimal edge detection algorithm. In this situation, an "optimal" edge detector means:

- *good detection* – the algorithm should mark as many real edges in the image as possible.

- *good localization* – edges marked should be as close as possible to the edge in the real image.
- *minimal response* – a given edge in the image should only be marked once, and where possible, image noise should not create false edges.

To satisfy these requirements Canny used the calculus of variations – a technique which finds the function which optimizes a given functional. The optimal function in Canny's detector is described by the sum of four exponential terms, but can be approximated by the first derivative of a Gaussian.

III. PARALLEL FUZZY C MEANS SEGMENTATION

The goal of image segmentation is to cluster pixels into salient image regions, i.e., regions corresponding to individual surfaces, objects, or natural parts of objects. Segmentation could be used for object recognition, occlusion boundary estimation within motion or stereo systems, image compression, image editing, or image database look-up.

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

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). Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture. Adjacent regions are significantly different with respect to the same characteristic(s).

Image segmentation is the primary step in image analysis, which is used to separate the input image into meaningful regions. There are two classifications exist to recognize a pattern, and they are supervised classification and unsupervised classification. A commonly used unsupervised classification method is a Fuzzy C Means algorithm [4]-[5]. Clustering is a process of partitioning or grouping a given sector unlabeled pattern into a number of clusters such that similar patterns are assigned to a group, which is considered as a cluster.

There are two main approaches to clustering which is crisp clustering and fuzzy clustering techniques. One of the characteristic of crisp clustering method is that the boundary between clusters is fully defined but in many real cases the boundary between clusters cannot be clearly defined. Some patterns may belong to more than one cluster. In such cases, the fuzzy clustering method provides a better and more useful method to classify these patterns.

Fuzzy clustering method and its derivatives have been used for pattern recognition, classification, data mining, and image segmentation. It has also been used for medical

image data analysis and modeling etc. Clustering is used for pattern recognition in image processing, and usually requires a high volume of computation. This high volume computation requires considerable amount of memory which may lead to frequent disk access, making the process inefficient. With the development of affordable high performance parallel systems, parallel algorithms may be utilized to improve performance and efficiency of such tasks.

The computation speed and memory requirement needed for executing FCM is a big hurdle which tried to overcome in this paper. In FCM, the cluster center initialized by random numbers and it requires more number of iteration for converging to a final actual cluster center. Due to the above fact that FCM consumes high iteration and also more sensitive to the initialization of cluster centers. Large and high resolution pictures such as medical images like MRI, it is not possible to accommodate the entire membership matrix in the available memory of the system. This leads to swapping of memory contents i.e. frequent disc access required. Improving the performance of the FCM based image segmentation by distributing computation and main memory usage.

The parallel FCM algorithm divides the image pixels equally among the processors so that each processor handles n/p data points (n is the number of pixels and p is the number of processors involved in the computation) [2]. The fuzzy membership function divided among the processors and is used for the calculation of the degree of membership, only for the local dataset. By dividing the data set among processors the likelihood of carrying out the computation only on the main memory of the processors and without the need to access the secondary storage. In this paper, the proposed algorithm is to simplify the dataset and find an initial cluster centers as close as possible to the actual cluster centers. This will reduce the number of iterations and also improve the execution performance. The initial cluster center found by the proposed algorithm approximates the actual cluster center very well. This enhances the performance and efficiency as compared to FCM algorithm.

A. Implementation of HSPFCM Algorithm

We proposed a High Speed parallel Fuzzy C Mean Algorithm (HSPFCM) algorithm to improve the computation time of MRI image segmentation based on the Fuzzy C Means algorithms. The theoretical background and the mathematical analysis were reported by many researchers in this field [1-2]. By applying the parallel processing concept and refining the initial cluster center, the proposed algorithm increases the speed of computation and requires less memory. The proposed HSPFCM is divided into two phases, they are

1. Cluster Center Algorithm (CCA)
2. Parallel Fuzzy C Mean (pFCM)

Cluster Center Algorithm (CCA)

First, we partition of dataset into several unit block cells by using the k-d tree method. For each unit block, we calculate the centroid of patterns in the unit block. The centroid of patterns will be used to represent all patterns in this unit block thereby a dataset X_n can be drastically

reduced to a simplified Xps containing the centroid of the original patterns. We then find the actual cluster center of this simplified dataset by the FCM algorithm.

The output of FCM algorithm is the initial cluster center for our algorithm. Selection of the initial cluster centers that approximates the actual cluster centers can reduce the number of iterations. Using the target tracking by initializing, every iteration in the procedure with the clustering result from the previous one can speed up the convergence significantly. Since the number of iterations required in the FCM algorithm strongly depends on the initial cluster centers, the goal of the proposed method is to simplify the dataset and find an initial candidate set of cluster centers as close as possible to the actual centers. This will reduce the number of iterations and improve the execution performance.

The CCA algorithm can be divided into different steps.

Step 1: First, we calculate scattered degree which is defined as the distribution range and the standard deviation of the feature partition the dataset into unit blocks by using the k-d tree method. The partitioning depends on the scattered degree of data values for each dimension. If one dimension has a higher scattered degree it has a higher priority to be split.

$$S_i = (X_{\max} - X_{\min})_i / \sigma_i \quad i = 1, \dots, f(1)$$

where, S_i is the scattered degree for the i th feature, X_{\max} is the maximum value of pattern in the i th feature, X_{\min} is the minimum value of pattern in the i th feature, σ_i is the standard deviation of all patterns in the i th feature and f is the number of features(dimensions).

Step 2: The k-d tree is used to split the dataset into unit block, which is a kind of binary partition based on the difference between the maximum and minimum values of the partition dimension. Thus, it may easily acquire the range of each block in each dimension. If the number of splits is p then $N_{ub} \leq 2^p$, where N_{ub} is number of unit block (number of splits in the k-d tree method).

Unit block depends on the total number of patterns N and the distribution of the dataset. There must be at least one pattern in each unit block. If there is no sample pattern in a unit block, the unit block will then be discarded.

Step 3: After constructing k-d tree, we calculate the centroid X_i for every unit block that contains some sample patterns. Then, we use all of these centroids X_i to denote the original dataset. The formula of calculating the centroid in the i th unit block is as follows:

$$X_i = LSUB_i / WUB_i \quad (2)$$

In addition, each centroid contains statistical information of the patterns in each unit block. These include the number of patterns in a unit block (WUB) and the linear sum of all patterns in a unit block (LSUB). When we scan the database the second time, it also finds the statistics of each dimension. These statistics will be used when the algorithm calculates new candidate cluster centers, which improves the system performance.

Step 4: Initialize the cluster center matrix by using a random generator from the dataset, record the cluster centers, and set $k = 0$.

Step 5: Initialize the membership matrix $M(k)$ by using the equation (3) with reduced data set.

Step 6: Increase k ($k = k+1$), compute the new cluster center matrix $C(k) = [c_j]$ with $M(k)$

Step 7: Compute the new membership matrix $M(k+1)$

Step 8: If $\|M(k+1) - M(k)\| < \epsilon$ then STOP; otherwise return to step 6.

Step 9: The output of Cluster Center Algorithm is set of cluster centers that will closet to the actual cluster center of FCM & that will apply as the input to the PFCM. The number of iterations needed to converge by CCA algorithm is also less than the number of convergence iterations of the FCM algorithm.

From the experimental results, it is shown that this algorithm improves the speed. For large datasets, the CCA algorithm improves the performance even more. For a fair comparison, the initialization in Phase I of the CCA algorithm and the FCM algorithm is determined randomly. It also found in the CCA algorithm that an initial cluster center selected from a unit block with a higher density is closer to the actual cluster center.

Parallel Fuzzy C-Mean (PfcM)

This algorithm enhances the segmentation of image by distributing computation and main memory usage. This algorithm converts image into pixel values (n), then the pixel values are equally divided and distributed to all processors so that each processor has n/p number of pixels (n is total number of pixel, p is a total number of processor) for execution [2]. The membership function is provided for each processor to calculate the degree of membership function, $\mu_{ci}(x_l)$. This algorithm avoids the use of external storage device because each processor uses its own main memory.

PFCM Algorithm

Given (X, C, w, ϵ) , where $X = \{X_0, X_1, \dots, X_{n-1}\}$ is a pixel value of the given image, $C = \{C_0, C_1, \dots, C_{k-1}\}$ is set of k clusters, w is weight that determines the degree to which partial members of a cluster affect the clustering results, and ϵ indicates the precision of results, cluster centre (c_i), such as equation 4 in CCA algorithm. The parallel processing consists of two main steps.

On the parent processor

1. Divide X into n/p subsets and send each subset to a participating processor using a one to all personalized broadcast.
2. Run CCA algorithm.
3. One to all broadcast cluster centers (c) which is calculated by CCA.

This algorithm is to divide the pixel value (n) horizontally (row wise order) and send among all processors (p), if a row of the pixel value is not a integral multiple of the number of processor, the remaining rows are divided among the processor from the lowest ID. For example, if the image pixel size is 256 and we consider three processors, so each processor will get 85 rows and 1 row is left, so the first processor will get one more row. After the completion of the computation of the centers of the clusters and final degree of membership matrix, accordance to the parallel algorithm discussed above, each processor defuzzifies its local data and sends it back to the initiating processor.

The parent processor, in turn, writes the received data into the output file to form the segmented image. The results obtained by applying FCM on a grey scale image file need to be defuzzified to obtain the resultant output image

file. The defuzzification process begins by transforming the final membership function matrix back to image by deciding to which cluster each pixel belongs. The decision on which cluster a pixel belongs to is made based on the winning, i.e. selecting the cluster for which the pixel has the highest degree of membership. If a pixel belongs to ci , then it will be painted with the color code of cluster i ; the resulting image will be partitioned by color-coded regions. This reversed mapping process forms the defuzzification task.

IV. DECISION TREE CLASSIFICATION

A decision tree is a type of multistage classifier that can be applied to a single image or a stack of images. It is made up of a series of binary decisions that are used to determine the correct category for each pixel. The decisions can be based on any available characteristic of the dataset. For example, you may have an elevation image and two different multispectral images collected at different times, and any of those images can contribute to decisions within the same tree. No single decision in the tree performs the complete segmentation of the image into classes. Instead, each decision divides the data into one of two possible classes or groups of classes.

A. Sample tree construction

The structure of a decision tree is primarily dependant upon which variable (in our case, which relative voxel) is interrogated at each tree node. Much of the literature on decision tree classifiers is concerned with methods of constructing trees which optimize certain properties, the most common objective being to minimize the expected traversal cost. For even a modest number of classes, there are a huge number of possible tree structures, so it is infeasible to simply generate and compare all alternatives.

There are construction methods which are known to minimize the expected testing cost, such as dynamic programming. All truly optimal methods, however, are computationally expensive, and have indeed been shown to be NP-complete. Moret [24] has identified two classes of heuristic that are used in sub-optimal decision tree generation. In the information heuristic, a node variable is chosen that maximizes the amount of additional information that is gained at that stage along the path traversal. In the splitting heuristic, a variable is chosen which divides the remaining class set into 2 disjoint sets which are effectively discriminated by the node variable.

V. ASSOCIATION RULE MINING

One of the most important Knowledge Distributed Database research issues is association rules. ARM task is to discover the hidden association relationship between the different itemsets in transaction database.

A. Frequent Itemsets and Maximum Frequent Itemsets

Let $I = \{i_1, i_2, \dots, i_m\}$ be a set of m distinct items. A transaction T is defined as any subset of items in I . A transaction

database T is said to support an itemset x contains in I if it contains all items of x . The fraction of the transaction in D that support x is called support value is above some user defined minimum support threshold then the itemset is frequent, otherwise it is infrequent. Maximum frequent itemsets have been denoted as F if all superset of frequent itemsets is infrequent itemsets. The maximum frequent itemsets that are discovered have been stored in the maximum frequent itemsets. Maximum frequent candidate set, which is the smallest itemsets, it includes all current frequent itemsets known, but it does not include any infrequent itemsets. The identification of maximum frequent itemsets in earlier stage can reduce the number of candidate itemsets generated. So that it will reduce the CPU and I/O time. In case if the maximum frequent itemsets discovered are long, then the performance of the algorithm will be excellent, so the issue of discovering frequent itemsets can be converted into the issue of discovering maximum frequent itemsets. All the maximum frequent itemsets identified has been stored in the transactional database.

B. Frequent Pattern Tree

The special prefix tree called FP-Tree consists of a frequent item header table. The root is labeled as "NULL" and a set of the item prefix subtree are called the children. There are four fields defined for each item prefix subtree, they are node-name, node-count, node-link node-count and node-parent [16], [17]. There are also three fields are included for each entry of the frequent item header table, those are item-name, item-sup, item-head. The pointer pointing to the first node in the FP-Tree carrying node name is item-head algorithm. The FP-Tree construction steps includes,

First TDB Scan:

Let L be the set of frequent items and support value of each frequent item have been collected.

(i) Scan TDB again:

Let L be the set of frequent items and support value of each frequent item have been collected.

(ii) Scan TDB again:

All frequent items in every transaction has been selected and sorted according to the order of list of frequent items. From the corresponding itemsets by combining the ordered frequent items in every transaction and insert them to FP-Tree respectively

C. Maximum Frequent itemsets mining

Definition 1: If an itemset is infrequent, all its superset must be infrequent.

Definition 2: If an itemset is frequent, all its subsets must be frequent.

Definition 3: If $A = \{a_1, a_2, \dots, a_{m-1}\}$ is an infrequent itemset, and $a_i = \{a_j / a_i \square A, j \neq i\}$ may be frequent itemset.

The top-down search algorithm uses the concept of definition 2, which begins from the only itemsets, it reduces some candidate itemsets in every pass. According to the definition 2, if K -itemsets is infrequent in K pass, then check all $(K-1)$ itemsets in next pass consequently it can discover all maximum frequent itemsets. The definition 4.1 can be used in the situation where maximum frequent itemsets are short. Hence in the proposed system, topdown search tactic is used which uses the maximum frequent candidate set to store the information. The support value of each maximum frequent itemsets has been counted respectively.

Maximum Frequent Itemset Finding Algorithm:

Input: FP-tree generated from transaction database; frequent item header table and the minimum threshold support (minsup) defined by user; list of frequent itemsets $LF = \{1,2,3,\dots,k\}$

Output: Maximum frequent itemsets (MFI)

Output: Maximum frequent itemsets (MFI)

- (1) For ($i=1; i \leq \max_level; i++$) // Where I is a variant which represents corresponding level in the hierarchical association rules.
- (2) $MFI_i = \phi$;
- (3) $MFC = FL = \{1,2,3,\dots,K\}$; // MFC is the maximum frequent candidate sets in Transactional database.
- (4) While ($MFC \neq \phi$) do begin
- (5) for ($j=k; j>0; j--$) do begin
- (6) $MFC_j = \{ c / C \leq MFC \text{ and item is the last item in } c \}$;
- (7) $MFC = MFC - MFC_j$;
- (8) Call Compute Count (FP_tree, frequent item header table , MFC);
- (9) for all $n \in MFC_j$ do begin
- (10) if $n.support \geq minsup$ then
- (11) $MFI_i = MFI_i \cup n$
- (12) Else
- (13) For all item $m \in n$ do

VI. CONCLUSION

A multiple level image mining technique for brain tumor classification using FP-Tree and decision tree classifier has been developed and the performance is evaluated. The proposed algorithm has been found to be performing well compared to the existing classifiers. The accuracy of 93% and sensitivity of 96% were found in classification of brain tumors. The developed brain tumor classification system is expected to provide valuable diagnosis techniques for the physicians.

ACKNOWLEDGEMENT

The authors would like to express their gratitude to Dr. D. Elangovan, Pandima CT scan centre, Dindigul for providing the necessary images for this study.

REFERENCES

[1] X.R. Marcela, H.B. Pedro, T.J. Caetano, M.A.M. Paulo, A.R. Natalia, and J.M. Agma , "Supporting Content-Based Image Retrieval and Computer-Aided Diagnosis Systems with Association Rule-Based Techniques," Data & Knowledge Engineering, in press.

[2] G.D. Tourassi, "Journey toward Computer-Aided diagnosis role of image texture analysis," Radiology, 1999, pp 317-320.

[3] S.R. Daniela, "Mining Knowledge in Computer Tomography Image Databases," Multimedia Data Mining and Knowledge Discovery, Springer London, 2007.

[4] B. Thomas, "Analyzing and mining image databases. Reviews Drug Discovery Today," Biosilico 10(11), 2005, pp.795-802.

[5] C. Ordonez, E. Omiecinski, "Image mining: A new approach for data mining," Technical Report GITCC-98-12, Georgia Institute of Technology, College of Computing, 1998, pp 1-21.

[6] H. Wynne, L.L. Mong, and J. Zhang, "Image mining: trends and developments. Journal of Intelligent Information Systems," 19 (1): 2002, pp 7-23.

[7] P. Stanchev , M. Flint, "Using Image Mining For Image Retrieval," In. Proc: IASTED conf. Computer Science and Technology, 2003, pp. 214-218.

[8] C. Ordonez, E. Omiecinski, "Discovering association rules based on image content," In Proc: IEEE Forum ADL, 1999, pp. 38-49.

[9] V. Megalooikonomou, J. For ,F. Makedon , "Data Mining in Brain Imaging," Statistical Methods in Medical Research, 2000, pp. 359-394.

[10] H. Pan, L.Jianzhong, W.Zhang, "Incorporating domain knowledge into medical image clustering," Applied Mathematics and Computation, 2007, pp. 844-856.

[11] J.B.W. Pluim, J.B.A. Maintz, and M.A. Viergever, "Mutual-informationbased registration of medical images," A survey. IEEE Transactions on Medical Imaging, 22(8):2003, pp. 986-1004.

[12] M.L. Antonie, O.R. Zaiane, and A.Coman, "Associative classifiers for medical images," Revised Papers from MDM/KDD and PAKDD/KDMCD, 2002, 68-83.

[13] B.A. Dogu, H. Markus, A.Tuukka, D. Prasun, and H. Jari , "Texture Based Classification and Segmentation of Tissues Using DT-CWT Feature Extraction Methods," In Proc: 21st IEEE International Symposium on Computer-Based Medical Systems, 2008, pp.614-619.

[14] P. Dollar, T. Zhuowen, T. Hai, and S. Belongie, " Feature Mining for Image Classification," In Proc: IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1-6.

[15] J. Dougherty, R. Kohavi, and M. sahami, "Supervised and Unsupervised Discretization of continuous features," In Proc: 12th International Conference Machine Learning, 1995, pp.56-69.

[16] R. Agrawal, T. Imielinski, And A.N. Swami, "Mining association rules between sets of items in large databases," In Proc: ACM SIGMOD Int. Conf. Manage, Washington, DC, 1993, pp. 207-216.

[17] S. Kotsiantis, D.Kanellopoulos, "Association Rules Mining: A Recent Overview," GESTS International Transactions on Computer Science and Engineering, 32 (1):2006, pp. 71-82.

[18] R. Agrawal, R.Srikant, "Fast algorithms for mining association rules," In Proc: Int. Conf. VLDB, Santiago, Chile, 1994, pp. 487-499.

[19] J.C. Felipe, A.J.M Traina, and C. Traina, "Retrieval by content of medical images using texture for tissue identification," In Proc: 16th IEEE Symp. Computer-Based Med. Systems. CBMS 2003, pp. 175-180.

[20] R. Abraham, J.B. Simha, and S.S. Iyengar, "Medical datamining with a new algorithm for Feature Selection and Naive Bayesian classifier," In Proc: 10th International Conference on Information Technology (ICIT), 2007, pp. 44-49.