DE GRUYTER OPEN

JAISCR, 2018, Vol. 8, No. 2, pp. 111 – 120
10.1515/jaiscr-2018-0008

TEXTURE AND GENE EXPRESSION ANALYSIS OF THE MRI BRAIN IN DETECTION OF ALZHEIMER'S DISEASE

Alhadi Bustamam, Devvi Sarwinda, Gianinna Ardenaswari

Department of Mathematics, Faculty of Mathematics and Natural Science, Universitas Indonesia
Kampus UI Depok, Indonesia 16424
devvi@sci.ui.ac.id

Submitted: 21st February 2017; Accepted: 27th March 2017

Abstract

Alzheimer's disease is a type of dementia that can cause problems with human memory, thinking and behavior. This disease causes cell death and nerve tissue damage in the brain. The brain damage can be detected using brain volume, whole brain form, and genetic testing. In this research, we propose texture analysis of the brain and genomic analysis to detect Alzheimer's disease. 3D MRI images were chosen to analyze the texture of the brain, and microarray data were chosen to analyze gene expression. We classified Alzheimer's disease into three types: Alzheimer's, Mild Cognitive Impairment (MCI), and Normal. In this study, texture analysis was carried out by using the Advanced Local Binary Pattern (ALBP) and the Gray Level Co-occurrence Matrix (GLCM). We also propose the bi-clustering method to analyze microarray data. The experimental results from texture analysis show that ALBP had better performance than GLCM in classification of Alzheimer's disease. The ALBP method achieved an average value of accuracy of between 75% - 100% for binary classification of the whole brain data. Furthermore, Bi-clustering method with microarray data shows good performance gene expression, where this information show influence Alzheimer's disease with total of bi-cluster is 6.

Keywords: Alzheimer's Disease, MRI, Feature Extraction, Bi-Clustering, Local Binary Pattern (LBP)

1 Introduction

Alzheimer's is a disease that causes damage to nerve tissue, so the memory cannot work well. Texture analysis is one approach that can be used to diagnose Alzheimer's disease and Mild Cognitive Impairment (MCI). Several texture approaches have been introduced to classify Alzheimer's, such as the Gray Level Co-occurrence Matrix (GLCM), Wavelet Transformation, the Statistical Approach, and Local Binary Pattern (LBP). A texture with high correlation can be seen in hippocampal volume changes at Alzheimer's [1]. Reference [2] shows the image texture analysis on MRI brain using statistical feature extraction approach, namely Gray Level Co-occurrence Matrix (GLCM) and wavelet transform. Whereas in other studies, 3D MRI brain images are used to detect between AD and MCI by looking at the texture of hippocampus [3]. There are also studies that suggest making the detection of AD and normal for 3D MRI brain images using GLCM approach with different size Region of Interest (ROI) [4]. In another study, the use of texture hippocampus to detect Alzheimer's and normal brain MRI of 2D images using statistical methods and wavelet transform was introduced [5]. In the same year, research of texture analysis usage, hippocampus volume and medial temporal lobe in AD patients by using GLCM approach to analyze texture were conducted [6]. The GLCM and Run

Length Matrix (RLM) have been used to analyze the texture of hippocampus. A further local statistical method based on co-occurrence matrix texture map is introduced to diagnose the onset of Alzheimer's by classifying between MCI and normal [7]. Result showed that this method outperformed gray matter local density method. The use of 3D Local Binary Pattern (LBP) in diagnosing Alzheimer's disease with FDG-PET images has also been introduced [8]. The proposed LBP is based on its volume.

The approach proposed by the authors was to use LBP for feature texture extraction. LBP has demonstrated its simplicity in computing. Basically, a LBP is formed based on the original sign value between the center pixel values and pixel values in the surrounding neighborhood. In this study, the expansion of the Local Binary Pattern (LBP) with the magnitude values are also used. This paper investigates the use of the sign value and magnitude of each joint Local Binary Pattern (LBP) that is formed for a 2D brain image. The proposed approach in this study consists of three stages, namely feature extraction, feature selection and classification.

In addition, genomic analysis using gene expression data is also used to find out where these genes influence risk for Alzheimer's. Several methods have been proposed previously.

In this study, we propose two methods for detection of Alzheimer's disease. As the first method, we used the texture analysis approach based on Local Binary Pattern (LBP) using sign and magnitude values that is called the Advance Local Binary Pattern (ALBP) for analyzing brain images. As the second method, we employed the genomic analysis approach using bi-clustering in microarray data.

The goal of this paper is to detect Alzheimer's disease, MCI, and Normal using texture and genomic analysis. Texture features analysis is a powerful quantitative approach for analyzing features in MRI data. An important property of texture analysis that makes it advantageous for use in MRI analysis is its robustness with respect to acquisition parameters. On the other hand, genomic analysis aims to discover how the genes influence risk for Alzheimer's.

2 Method

To analysis the performance of the proposed approach in this research, we conducted two kinds of analysis. The first was a texture analysis of brain images, and the second was analysis of gene expression using microarray data. The research workflow can be seen in Figure 1.

Figure 1. Research Workflow

2.1 Texture Analysis of Brain Images

In this study, the proposed technique consisted of three stages, namely feature extraction, feature selection, and classification.

Feature Extraction using the Advanced Local Binary Pattern

In feature extraction scheme, extension of LBP was used. This approach utilizes LBP with the use of sign and magnitude value, where pixel values are located at the center of the neighboring values [9]. An LBP with a sign value of the original LBP was introduced by [10]. In this paper, we called this approach as the Advanced Local Binary Pattern of Sign (ALBPS). ALBPS is basically obtained from binary process derivatives with the following formula

$$ALBPS_{P,R} = \sum_{p=0}^{P-1} sign(g_p - g_c)2^p \qquad (1)$$

and

$$sign(x) = \begin{cases} 1, & x \geq c \\ 0, & x < c \end{cases}, \tag{2}$$

where P is the number of neighbors and R is the radius.

Meanwhile, the LBP that utilizes the magnitude value called the Advanced Local Binary Pattern of Magnitude (ALBPM) is formulated as follows

$$ALBPM_{P,R} = \sum_{p=0}^{P-1} t(m_p - c)2^p \tag{3}$$

and

$$t(x,c) = \begin{cases} 1, & x \geq c \\ 0, & x < c \end{cases}, \tag{4}$$

where c is the average number of magnitude from whole images.

Both ALBPS and ALBPM produce binary strings, so they can be used simultaneously for feature extraction. Furthermore, the code of ALBPS and ALBPM can be used to produce a joint 2D histogram. The ALBP scheme is represented as "ALBPSM". This method is used for feature extraction of 2D multi slice images.

In this research, the authors propose an approach to feature extraction of 3D volume data. This approach begins with the use of three planes of a volume image, namely *xy, xz* and *yz*. In a previous study, this approach was introduced with the name of the Local Binary Patterns from Three Orthogonal Planes (LBP-TOP) [11]. LBP-TOP apply LBP uniform with sign values for each plane. Meanwhile, the authors use the ALBPSM of each plane as a 3D descriptor in this study. The 3D image is converted into 2D images and then *xy, xz* and *yz* are obtained for each plane. Furthermore, ALBPSM is calculated for all slices in each plane in order to obtain the ALBPSM histogram for planes *xy, xz* and *yz*. Afterwards, the histograms of the three planes are combined into one, and it is used as the input feature vectors. We represent this method as an Advanced Local Binary Pattern of Sign and Magnitude from Three Orthogonal Planes (ALBPSM-TOP). ALBPSM-TOP also implemented the utilizing rotation invariant and uniform approach [12, 13]. An illustration of ALBPSM-TOP can be seen in Figure 2.

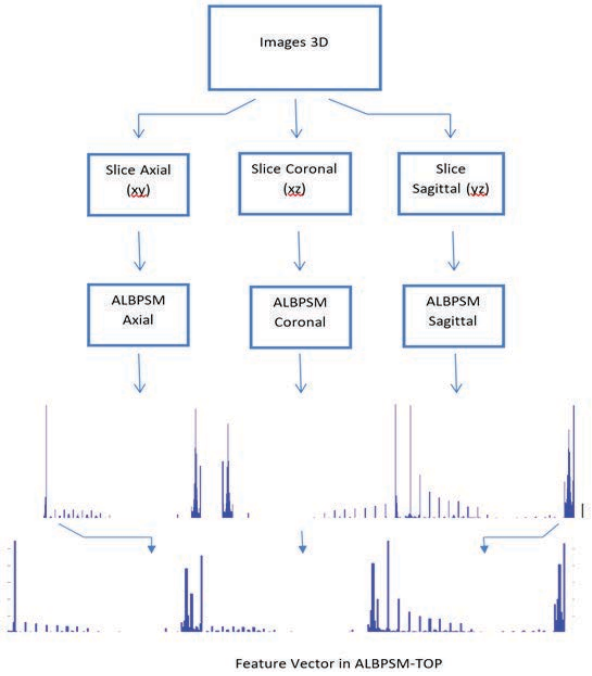

Figure 2. Scheme of ALBPSM-TOP Method

Uniform Local Binary Pattern and Rotation Invariant Local Binary Pattern

An LBP is called uniform if the binary vector T contains at most two transitions 0 and 1 or vice versa when the bit pattern is considered circular. Figure 2 highlights the pattern shapes that are uniform and not uniform. Another approach proposed such as LBP uniform and LBP rotation invariant [14]. Unlike the LBP uniform, the LBP rotation invariant is trying to do the rotations of a textured input image because the LBP patterns are translated into different locations and rotated around their origin.

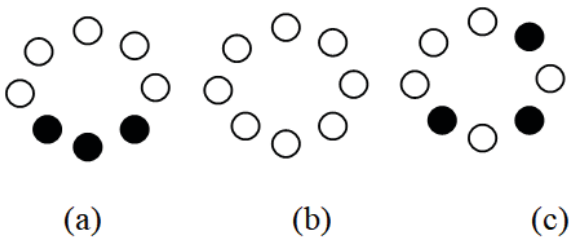

Figure 3. Three examples of LBP. Black express binary values 0, and 1 of a binary value of the white states. Shape (a) is LBP uniform since it has two transitions. Shape (b) is LBP uniform since it has no transition. LBP (c) is not uniform since it has six transitions.

We construct the histogram by using normalization values of LBP codes obtained from rotation invariant mapping. In this mapping, each binary value of the LBP code is circularly rotated into the minimum value with the following formula

$$LBP^{ri}_{P,R} = \min_i ROR(LBP_{P,R}, i), \quad (5)$$

where *ROR* (*x, i*) is the circular bitwise right rotation of bit sequence of *x* by *i* steps. The histogram of $LBP^{ri}_{P,R}$called invariant if the rotation angle of input image is$\alpha = x\frac{360^\circ}{P}$, $x = 1,2,...,P-1$. This method is robust for analyzing brain images with different angles [4].

Feature Selection using Factor Analysis

The purpose of factor analysis (FA) is to reduce the redundancy among variables using the smallest number of factors. FA correlates more between variables that are used for a specific factor. FA models can be simply defined as follows:

Suppose there is a set of random N observable random variables $x_1, x_2, x_3, ..., x_n$ with means$\mu_1, \mu_2, \mu_3, ..., \mu_n$. Then, suppose that there are several unknown constants; $\lambda_{i,j}$ is factor loading and m is the number of unobserved random variables from$f_j$factor, with $i \in 1, ..., N$, and $j \in 1, ..., m$ where $m < N$.

where x is a vector of the observed variable,$\mu$is the constant vector of mean,$\Lambda$is a constant N-by-m matrix of factor loadings, $\boldsymbol{f}$ is a common factors matrix and z is an error vector.

$$x_i = \mu_i + \lambda_{i1} f_1 + ... + \lambda_{im} f_m + \varepsilon_i, \quad (6)$$

where$\varepsilon_i$is i-th error value. If written in a matrix form, it will result the following formula:

$$x_i = \mu + \Lambda f + \varepsilon, \quad (7)$$

where x is a vector of the observed variable,$\mu$is constant vector of mean,$\Lambda$is constants N-by-m matrix of factor loadings, $\boldsymbol{f}$ is a common factors matrix and z error vector.

The following assumption is applied to the unobserved random matrix $\boldsymbol{f}$, where there is no correlation between $\boldsymbol{f}$ and z. The mean of $\boldsymbol{f}$ is equal to 0 and the covariance of $\boldsymbol{f}$ is the identify matrix. This assumption allows computing the factor loadingsΛusing the maximum likelihood approach [15]. Once the factor loadings have been estimated, then this result will be used as a feature vector for classification purposes.

Classification

In this study, Support Vector Machine and the Nave Bayes Classifier are used for classification because of their ability to pattern the classification. Assume that we want to construct a linear classifier that can correctly separate the two populations in a high-dimensional space. Performance of the classification is influenced by the model used for classification. The authors used k-fold cross validation with k = 10 as a model for learning performance data of the training and testing data.

To validate the effectiveness of our proposed method, the value of accuracy, sensitivity and specificity are measured with the following equation

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

$$Accuracy = \frac{TP}{TP + FN} x100\% \quad (9)$$

$$Accuracy = \frac{TN}{TN + FP} x100\% \quad (10)$$

In addition to evaluating the accuracy, kappa statistics are also used to measure the agreement between predicted and observed categorizations of a dataset, while correcting for an agreement that occurs by chance.

2.2 Gene Expression Analysis of Microarray Data

One of the usual goals in expression data analysis is to group genes according to their expression under multiple conditions, or to group conditions based on the expression matrix. In gene expression, data clustering can be done with a bi-cluster algorithm, which is a clustering method that involves not only the object to be clustered, but also the properties or condition of the object.

In this study, the proposed technique is two-phase bi-clustering method. The first phase is implementation of parallel k-means to generate k clusters. Furthermore, the H-score of the clusters are calculated and checked to see whether they are within the threshold value. If the cluster's H-score

is within the threshold value, then we get the resultant bi-cluster with all the condition (columns). However, with all the conditions, the clusters that are generated may have Hscore values greater than the threshold. So the second phase of the algorithm is required where the residue score of Cheng and Church is applied to each of the columns of the clusters.

Bi-clustering of expression data by Cheng and Church

Bi-clustering of expression data using the method of Cheng and Church involves an iterative greedy search. The greedy algorithm follows the strategy of making a local optimal choice in each step, in order to find a global optimum. This kind of heuristic does not ensure an optimal global solution will be obtained, but approximates it in a reasonable time. They work by either recursively or iteratively constructing a set of objects from the smallest possible constituent parts [16].

Cheng and Church were the first to apply bi-clustering to gene expression data. Their algorithm adopts a sequential covering strategy in order to return a list of n bi-clusters from an expression data matrix. In order to assess the quality of a bi-cluster, the algorithm makes use of *mean squared residue* (MSR). This measure aims at evaluating the coherence of the genes and conditions of a bi-cluster by using the means of genes and conditions expression values in it.

Let X be the set of genes and Y the set of conditions. Let a_{ij} be the element of the expression matrix A representing the logarithm of the relative abundance of the mRNA of the ith gene under the jth condition. Let $I \subset X$ and $J \subset Y$ be subsets of genes and conditions. The pair (I,J) specifies a submatrix A_{IJ} with the following MSRscore.

$$(I,J) = \frac{1}{|I|\,|J|} \sum_{i\in I, j\in J} (a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2, \quad (11)$$

where

$$a_{iJ} = \frac{1}{|J|} \sum_{j\in J} a_{ij}, a_{Ij} = \frac{1}{|I|} \sum_{i\in J} a_{ij}, \quad (12)$$

and

$$a_{IJ} = \frac{1}{|I|\,|J|} \sum_{i\in I, j\in J} a_{ij} = \frac{1}{|I|} \sum_{i\in I} a_{iJ} = \frac{1}{|J|} \sum_{j\in J} a_{Ij}, \quad (13)$$

are the row and column means and the mean in the submatrix (I,J). A submatrix A_{IJ} is called a δ - *bi-cluster if* $H(I,J) \leq \delta$ for some $\delta \geq 0$.

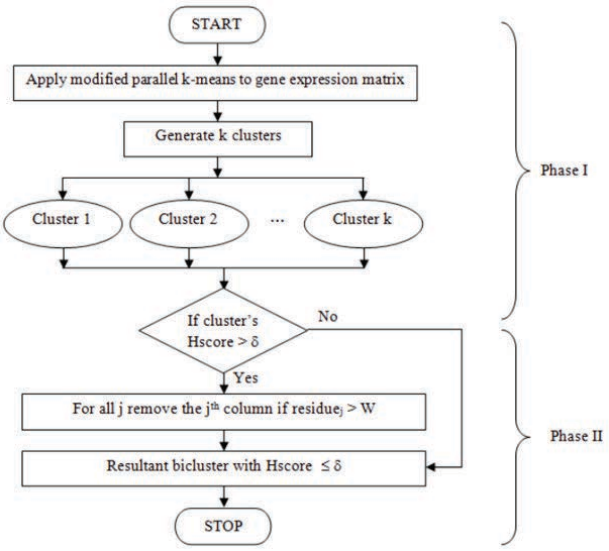

Figure 4. Bi-clustering Method Flowchart

Figure 5 shows a scheme of CC Biclustering Algorithm. The bi-clustering algorithm takes as input the expression matrix EM and the threshold δ imposed on MSR. δ is used to reject non δ-bi-clusters. A list L of δ-bi-clusters is returned as output. After pre-processing the missing values of the input data matrix by replacing them with random numbers, the bi-cluster discovery process is repeated as many times as the bi-clusters are desired. In each iteration, the bi-cluster B is initialized to the whole matrix. Next, three different phases for multiple node deletions, single node deletion, and node addition are applied. These phases iteratively perform the removal and addition of rows and columns, ensuring that the result is a δ-bi-cluster. Finally, a substitution phase replaces the elements of the input matrix that are contained in the recently found bi-cluster with random values. This substitution is applied in order to prevent overlapping among bi-clusters, since it very unlikely that elements covered by existing bi-clusters would contribute to any future bi-cluster. Although this strategy succeeds in avoiding overlapping, CC has several drawbacks due to element masking and also due to the use of a threshold for rejecting solutions, which is depen-

dent on each dataset and must be computed before applying the algorithm [17].

Algorithm 1. Cheng and Churchs algorithm

Input: Expression Matrix EM; Thresholds δ

Output: List of Biclusters L

1: Preprocess the missing values of EM
2: List $L = \oslash$
3: Bicluster B
4: **repeat** n times
5: $B = EM$
6: B_δ =multiple node deletion phase (B, δ)
7: B'_δ = simple node deletion phase (B_δ, δ)
8: B''_δ = addition phase (B'_δ, δ)
9: $L = L \oplus B''_\delta$
10: Substitution phase (B''_δ, EM)
12: **return** L

2.3 Data Acquisition

The data used in this research were of two kinds: brain images from MRI and gene data from microarray data. Brain MRI images were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database. The characteristics of the subjects used in this study are described in Table 1.

We also used data from a variety of 2D and 3D MRI scanners using the 1.5T system at T1. Gene data were obtained from the Allen Brain Atlas. The data consist of five samples or conditions and 1054 probe genes.

Table 1. Characteristics of the three subject groups: AD, MCI, and Normal

Attribute	AD	MCI	Normal
#subjects	41	45	52
Age	79.7(5.5)	77.8(5.9)	76.3(3.5)
Sex(%male)	53.6	63	45.3
MMSE	20.3 (3.8)	25.2(2.7)	29.7(0.9)
CDR	0.5 - 1.0	0.5	0

Note: Mean (Standard Deviation)

3 Result and Discussion

In this study, LBP utilize the number of neighbors and radius used were 16 and 2 respectively. Due to the selection of 2 radius and 16 neighbors, LBP showed good performance in comparison with one radius and eight neighbors [10]. From FA, we get 14 best features as input vector for classification. Finally, the classification was carried out using WEKA tools version 3.7.9. The authors chose SVM and the Nave Bayes function as classifiers. Furthermore, 10-fold cross validation was selected as a model for learning performance data.

The experimental results for texture analysis can be seen in Table 3. Table 3 shows the ALBPSM-TOP Uniform Rotation Invariant for 3D images has the highest average accuracy of 100% for Normal with AD and for Normal with MCI. However, AD with MCI gets 94.76% for accuracy value.

The ALBPSM-TOP Uniform Rotation Invariant method outperformed other methods.

From microarray data, the experimental results in Table 2 indicate there are six bi-clusters that show gene expression data that influence risk of Alzheimer's disease. The data consist of five samples or conditions and 1054 probe genes. After preprocessing, we deleted some null data, thus new data consisted of five samples/conditions and 841 probe genes. Furthermore, the result in Figure 6 shows that the bi-clustering algorithms can be used to diagnosis of clinical attributes for a clear understanding of patterns of doctors.

4 Conclusion

From our results, texture analysis and gene expression analysis using microarray data show good performance in detection of Alzheimer's disease. Our advanced local binary pattern method in detection of Alzheimer's disease was able to achieve accuracy, sensitivity and specificity above 95% for all evaluations. Furthermore, the bi-clustering method can show the gene expression data that influences risk in Alzheimer's disease.

Acknowledgement

The authors would like to thank for Alzheimer's Disease Neuroimaging Initiative (ADNI) database (www.loni/ucla.edu/ADNI) to provide and support for this research.

Table 2. List of Biclusters

Bicluster	Size	Probe Genes
1	5 11	X74, X118, X215, X282, X295, X495, X527, X534, X597, X751, X773
2	4 12	X1, X74, X118, X215, X282, X295, X495, X527, X534, X597, X751, X773
3	4 54	X49, X74, X89, X118, X137, X147, X167, X294, X205, X215, X225, X229, X232, X240, X266, X282, X287, X295, X459, X460, X463, X467, X469, X472, X486, X492, X495, X526, X527, X534, X547, X597, X605, X609, X612, X617, X626, X630, X664, X688, X711, X750, X751, X753, X757, X760, X762, X763, X773, X778, X791, X817, X840
4	4 12	X74, X118, X215, X282, X295, X495, X527, X534, X597, X676, X751, X773
5	4 38	X23, X74, X86, X118, X213, X214, X215, X216, X217, X223, X227, X270, X282, X295, X370, X371, X376, X400, X457, X461, X495, X506, X525, X527, X534, X546, X597, X673, X683, X687, X751, X765, X769, X771, X773, X776, X787, X818
6	4 14	X74, X102, X118, X215, X282, X295, X338, X347, X495, X527, X534, X597, X751, X773

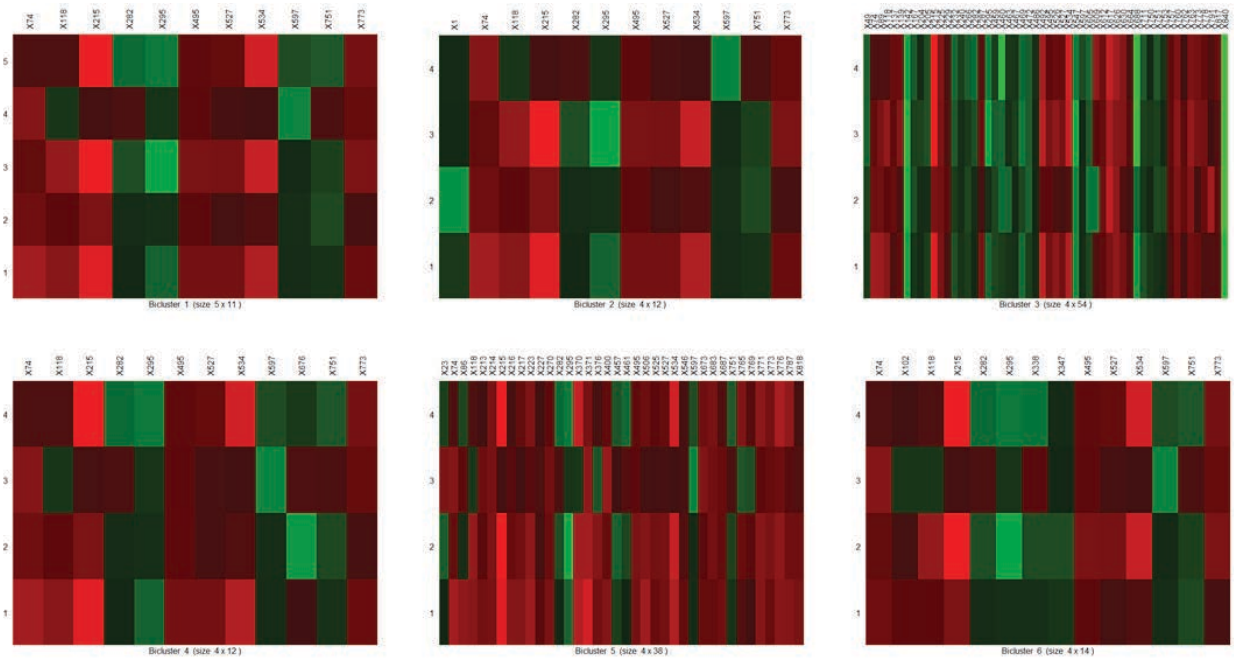

Figure 5. Result of heatmap of Biclusters

Table 3. Classification results using SVM and Naive Bayes for binary class

Method	Results	**Alzheimer vs Normal**		**MCI vs Normal**		**Alzheimer vs MCI**	
		SVM	NB	SVM	NB	SVM	NB
LBPTOP-*Uniform*	Accuracy	76.71	74.44	63.49	64.02	56.07	59.54
	Sensitivity	87.5	85	50.54	48.39	35.48	24.73
	Specificity	67.71	65.63	76.04	79.2	80	100
	Kappa	0.54	0.495	0.2268	0.277	0.15	0.233
ALBPS-TOP *Rotation Invariant*	Accuracy	99.47	99.47	98.3	98.3	72.25	72.25
	Sensitivity	98.92	98.92	98.75	98.75	80.64	80.64
	Specificity	100	100	97.92	97.92	62.5	62.5
	Kappa	0.989	0.989	0.966	0.966	0.436	0.436
ALBPS-TOP *Uniform Rotation Invariant*	Accuracy	**100**	**100**	**100**	**100**	78.61	78.61
	Sensitivity	**100**	**100**	**100**	**100**	80.64	80.64
	Specificity	**100**	**100**	**100**	**100**	76.25	76.25
	Kappa	**1**	**1**	**1**	**1**	0.569	0.569
ALBPSM-TOP *Uniform*	Accuracy	**100**	99.47	**100**	98.3	94.22	94.22
	Sensitivity	**100**	98.92	**100**	98.75	94.62	94.62
	Specificity	**100**	100	**100**	97.92	93.75	93.75
	Kappa	**1**	0.989	**1**	0.966	0.884	0.884
ALBPSM-TOP *Uniform Rotation Invariant*	Accuracy	**100**	98.94	**100**	98.86	**94.79**	94.22
	Sensitivity	**100**	100	**100**	100	**94.62**	94.62
	Specificity	**100**	97.92	**100**	97.92	**95**	93.75
	Kappa	**1**	0.978	**1**	0.977	**0.895**	0.884
GLCM	Accuracy	94.6	96.2	87.4	89.5	55.79	60.43
	Sensitivity	90.5	95.4	84.3	85.4	35.48	30.45
	Specificity	93.2	96.1	82.1	87.4	80	93
	Kappa	0.91	0.94	0.86	0.85	0.17	0.256

References

[1] Zhou X, Liu Z, Zhou Z, Xia H: Study on Texture Characteristics of Hippocampus in MR Images of Patients with Alzheimer's Disease. Proc. 3rd Annu. Conf. Biomedical Engineering and Informatics 2010, Yantai, Beijing.

[2] Kassner A and Thornhill R.E: Texture Analysis: A Review of Neurologic MR Imaging Application. American Journal of Neuroradiology 2010, 31: 809-816.

[3] X. Li, H. Xia, Z. Zhuo, L. Thong, 3D Texture Analysis of Hippocampus Based on MR Images in Patients with Alzheimer Disease, and Mild Cognitive Impairment," in International Conference on Biomedical Engineering and Informatics, Beijing, 2010.

[4] J. Zhang J, Y. Chunsui , and Gui Lian J, 3D texture analysis on MRI images of Alzheimer's disease, Brain Imaging and Behavior, vol. 6, pp. 61-69, 2012.

[5] Rajeesh J, S.M. Rama, Palinikumar S, Gopalakhrisnan T: Discrimination of Alzheimer's disease using hippocampus texture features from MRI. Journal Asian Biomedicine 2012, 6: 87-94.

[6] Xia H, Tong L, Zhou X, Zhang J: Texture Analysis and Volumetry of Hippocampus and Medial Temporal Lobe in Patients with Alzheimer's Disease. in International Conference on Biomedical Engineering 2012, Macau, Macao.

[7] Simões R, Slump C, Marie A: Using local texture maps of brain MR images to detect Mild Cognitive Impairment. 21st International Conference on Pattern Recognition 2012, Japan

[8] P. Morgado, M. Silveira, and J.S. Marques, J. Computer Methods in Biomechanics and Biomedical Engineering: 1, 183 (2013)

[9] Ojala T: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. Journal IEEE Transaction on Pattern Analysis and Machine Intelligence 2002, 24: 971-987.

[10] Pietikainen M, Zhao G, Hadid A, Ahonen T: Local Binary Patterns for Still Images. Computer Vision Using Local Binary Patterns. London: Springer ; 2011 13-37.

[11] Guo Z, Liu Z , D Zhang: A Completed Modeling of Local Binary Pattern Operator for Texture Classification. IEEE Transactions on Image Processing 2010, 19: 1657-1663.

[12] Unay D, Ekin A, Cetin M, Jasinchi R, Erchil A: Robustness of Local Binary Patterns in Brain MRI Analysis. in Proc. 29th Ann. Conference of the IEEE EMBS 2007, Lyon.

[13] D. Sarwinda and A. Bustamam, Detection of Alzheimer's disease using advanced local binary pattern from hippocampus and whole brain of MR images, 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 2016, pp. 5051-5056

[14] T. Ojala, Multiresolution gray-scale, and rotation invariant texture classification with local binary patterns, Pattern Analysis and Machine Intelligence, vol. 24, pp. 971-987, 2002.

[15] A. C. Rencher, Editor, Methods of Multivariate Analysis, 2nd ed, John Willey & Sons Publishers, Canada, 2002.

[16] T. Ahonen, J. Matas, C. He, and M. Pietikainen, Editors. Proceedings of the 16th Annual Scandinavian Conference on Image Analysis, (2009) June 15-18; Oslo, Norway.

[17] Nanni L, Lumini A, Brahnam S: Local Binary Pattern Variants as Texture Descriptors for Medical Image Analysis. Artificial Intelligence in Medicine 2010, 49: 117-125.

[18] Association A: 2012 Alzheimer's disease facts and figures. Alzheimer's and Dementia: The Journal of the Alzheimer's Association2012, 8:131-168.

[19] Ojala T, Pietikinen M, and Menp T: A comparative study of texture measures with classification based on featured distributions. Journal Pattern Recognition 1996, 29: 51-59.

[20] Ahonen T, Matas J, He C, Pietikainen M: Rotation Invariant Image Description with Local Binary Pattern Histogram Fourier Features. Proc. 16th Annual Scandinavian Conference on Image Analysis 2009, Norway.

[21] M. Das, B. Borah. Biclustering of Gene Expression Data Using Two-Phase Method. International Journal of Computer Applications Vol. 103 No. 13. 2014.

[22] H. Turner, T. Bailey, W. Krzanowski. Improved Biclustering of Microarray Data Demonstrated through Systematic Performance Tests. Elseiver. Computational Statistics & Data Analysis, pp. 235 – 254. 2005.

[23] T. Kanungo, D. Mount, N. Netanyahu, et al. An Efficient K-Means Clustering Algorithm: Analysis and Implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 24 (7), pp. 881 – 892. 2002.

[24] A. Bustamam, G. Ardaneswari, D. Lestari, H. Tasman. Performance Evaluation of Fast Smith-Waterman Algorithm for Sequence Database Searches using CUDA GPU-Based Parallel Computing. Journal of Next Generation Information Technology Vol. 5 No. 2, pp. 38 – 46. 2014.

[25] K.S. Pollard, M.J. Van de Laan. Statistical Inference for Simultaneous Clustering of Gene Expression Data. Math Biosci, 176, pp. 99 – 121. 2002.

[26] S.C. Mdaeira, A.L. Oliveira. Biclustering Algorithms for Biological Data Analysis: A Survey. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 1, pp. 24 – 45. 2004.

[27] L. Lazzeroni, A. Owen. Plaid Models for Gene Expression Data. Statistica Sinica 12, pp. 61 – 86. 2002.

[28] J.A. Hartingan. Clustering Algorithm. New York: John Willey and Sons, Inc. 1997.

[29] Zhang D, Wang Y, Zhuo L, Yuan H, Shen D: Multimodal Classification of Alzheimer's Disease and Mild Cognitive Impairment. Journal Neuroimage 2011, 5: 856-867.

Alhadi Bustamam received the BSc (honor) degree in computational mathematics in 1996 and the master's degree in computer science in 2002 from University of Indonesia. Furthermore, he received the PhD degree in bioinformatics from Institute for Molecular Bioscience, the University of Queensland, Australia, in 2011. He was a Chair of Indonesian Mathematical Society for West Java, Jakarta and Banten Sections for 2013-2015 and 2015-2017 periode. Currently, he is a Chair of Department of Mathematics, Universitas Indonesia for 2014-2018 periode. His research focuses on high-performance computing approaches and data analysis techniques on computational mathematics, computational biology, bioinformatics, biomedical informatics and computer science.

Devvi Sarwinda earned her master's degree in computer science from Universitas Indonesia in 2013. She completed her B.Sc. in mathematics at University of Riau in 2009. She was research fellow at the Pattern Recognition Laboratory, Universitas Indonesia and Rumah Sakit Cipto Mangun Kusumo Jakarta in 2013. In 2015, she join at Universitas Indonesia as a lecturer and young researcher of Department of Mathematics. Currently, her research focuses on Biomedical and Health Informatics, Computational Intelligence, and Machine Learning.

Gianinna Ardaneswari received her master's degree in Department of Mathematics from Universitas Indonesia in 2017. She completed her B.Sc. in mathematics at Universitas Indonesia. Currently, she is a lecturer in Department of Mathematics, Universitas Indonesia.