Indonesian Journal of Electrical Engineering and Computer Science
Vol. 40, No. 3, December 2025, pp. 1607~1617
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v40.i3.pp1607-1617

Mobile application for diagnosing alzheimer's based on clinical dementia rating

Retno Supriyanti[1], Muhammad Putra Yubiksana[1], Bintang Abelian Mahardika Wijonarko[1], Yogi Ramadhani[1], Muhammad Syaiful Aliim[1], Mohammad Irham Akbar[1], Haris Budi Widodo[2], Wahyu Widanarto[3], Muhammad Alqaaf[4]
[1]Department of Electrical Engineering, Faculty of Engineering, Jenderal Soedirman University, Purwokerto, Indonesia
[2]Department of Dentistry, Faculty of Medical, Jenderal Soedirman University, Purwokerto, Indonesia
[3]Department of Physics, Faculty of Mathematics and Natural Sciences, Jenderal Soedirman University, Purwokerto, Indonesia
[4]Graduate School of Science and Technology, Nara Institute of Science and Technology, Ikoma, Japan

Article Info

Article history:

Received Mar 12, 2025
Revised Oct 17, 2025
Accepted Nov 16, 2025

Keywords:

Alzeimer
Classification
Mobile application
MRI machine
Neurodegenerative

ABSTRACT

Alzheimer's is a neurodegenerative disease characterized by memory loss, impaired thinking abilities, and changes in behavior. It is the most common form of dementia, significantly affecting a person's ability to carry out daily activities. Statistics indicate that the number of individuals suffering from Alzheimer's worldwide continues to rise as the population ages. Diagnosing Alzheimer's is a complex process that typically requires a skilled medical team. One diagnostic tool that can be utilized is an MRI machine. Previous research focused on extracting features from MRI images taken from three different cross-sections: axial, coronal, and sagittal. Based on these three types of cross-sectional images, we developed a system to classify the severity of Alzheimer's. This paper focuses on creating an Alzheimer's classification system accessible through a mobile application. The results indicate that our system has a performance accuracy of 90% in classifying the severity of the disease.

Corresponding Author:

Retno Supriyanti
Department of Electrical Engineering, Faculty of Engineering, Jenderal Soedirman University
Purwokerto, Indonesia
Email: retno_supriyanti@unsoed.ac.id

1. INTRODUCTION

Alzheimer's disease is a degenerative brain disorder that results in memory loss, cognitive decline, and behavioral changes. This condition damages brain cells, leading to a gradual deterioration of mental functions [1]. Diagnosing Alzheimer's disease requires careful and thorough examination, as the early symptoms often resemble those of other health conditions. Brain scans, such as CT scans or MRIs, can help identify structural changes in the brain associated with Alzheimer's. Currently, there is no definitive cure for Alzheimer's disease, making early detection essential to slow the progression of the disease and improve the quality of life for those affected. A critical aspect of the diagnostic process is the use of brain scans, which can reveal the structural changes in the brain that support an Alzheimer's diagnosis. Researchers have conducted studies using image-processing techniques to assist in the diagnosis of Alzheimer's disease. Georges *et al.* [2] employed two methods for MRI diagnosis based on MRI images. The first method involves reducing the noise in the existing photos using a non-linear filter, followed by image segmentation with the K-Means algorithm. The second method reduces image features using principal component analysis (PCA), after which a neural network algorithm classifies the images. Dara *et al.* [3] analyzed 80 publications on machine learning for diagnosing Alzheimer's disease and recommended several warranting further

investigation. Chaudhary *et al.* [4] utilized convolutional neural networks (CNNs) to identify Alzheimer's disease. They analyzed the entire image using CNNs and various filters to create visual slices. The activation functions employed included commonly used options such as sigmoid, tanh, and the corrected linear unit (ReLU). Prasath and Sumathi [5] researched the necessity of medical image processing and analysis methods for non-invasive Alzheimer's diagnosis. Ali *et al.* [6] adjusted the Fuzzy K-means algorithm to segment brain tissue affected by Alzheimer's disease. Additionally, based on this segmentation, they classified Alzheimer's severity using a hybrid technique. Gayathri *et al.* [7] employ an integrated approach for MRI-based Alzheimer's detection. Neural networks are used for feature extraction, followed by classification optimized through spider monkey optimization (SMO) to enhance diagnosis results. Bamber and Vishvakarma [8] created a model to diagnose and monitor the progression of Alzheimer's disease using artificial neural network techniques. Aljabar *et al.* [9] used MATLAB to preprocess MRI images and then applied YOLOv2 for Alzheimer's diagnosis. Mahanty *et al.* [10] suggested combining ensemble methods with deep learning models to detect Alzheimer's disease through MRI scans. The Xception architecture was employed to create multiple snapshots that offer a diverse range of MRI features. Wang *et al.* [11] proposed the addition of a new multimodal feature called "p-magnetic resonance imaging (MRI) value" to enhance the construction of 3D fusion images.

Additionally, they developed a deep joint learning diagnostic model, which included training a residual network (ResNet) to identify local pathological features for diagnosing Alzheimer's disease. Saratxaga *et al.* [12] apply deep learning techniques for Alzheimer's diagnosis using images from the open access series of imaging studies (OASIS) database. Jahan *et al.* [13] proposed using multimodal datasets for Alzheimer's diagnosis by integrating clinical data, MRI images, and psychological information. Parvatham and Maguluri [14] proposed a hybrid model designed to process MRI images of the brain to detect dementia. This model consists of two components. The first component involves feature extraction through convolution operations, while the second component uses these extracted features to train a support vector machine (SVM) for binary classification. Arya *et al.* [15] proposed A systematic review was conducted on publications that employed machine learning methods for the early detection of Alzheimer's disease. The primary objective of this review was to compare the effectiveness of PET and MRI imaging modalities in identifying Alzheimer's and evaluate each method's performance. Pasnoori *et al.* [16] developed an adaptive multi-thresholding algorithm using smoothed morphological histograms to identify neurodegenerative diseases at varying levels: normal, very mild, mild, and moderate. Our work also focused on creating this adaptive multi-thresholding algorithm for the same purpose. Tian *et al.* [17] developed a degradation model for PET image reconstruction that concentrates on recovering both high and low frequencies. By combining multilevel residual connections, this model enhances image quality, which aids in diagnosing Alzheimer's disease and other neurological disorders of the brain. Sener *et al.* [18] utilize deep learning to classify patients with normal cognitive function and mild cognitive impairment, as well as to detect Alzheimer's at an early stage before it advances. Begum and Selvaraj [19] enhanced the fast region-based convolutional neural network (R-CNN) approach by incorporating Bayesian optimization in the brain image segmentation process, making it more effective for Alzheimer's diagnosis. Li *et al.* [20] The model they use is the enhanced residual attention network (ERAN), which can be used for medical image classification by combining residual learning, attention mechanisms, and soft thresholds to improve feature accuracy. Ghadami and Rahebi [21] utilized CNN and the gray level co-occurrence matrix (GLCM) to extract features from MRI images of Alzheimer's patients. Next, they implemented the Harris Hawks optimization (HHO) algorithm to identify the most significant features. After selecting these features, they trained a multi-layer perceptron (MLP) neural network and employed a long short-term memory (LSTM) network for classification. Supriyanti *et al.* [22] reviewed how machine learning detects Alzheimer's disease. Research indicates that nearly all existing studies rely on neural networks and desktop-based learning machines to detect Alzheimer's disease.

However, none of these studies have explored the use of mobile applications to enhance the mobility of doctors in diagnosing Alzheimer's. Our previous research [22]-[25]. We conducted research on the development of a system for detecting Alzheimer's disease. Our study began with feature extraction from the hippocampus and ventricles in each MRI image cross-section, specifically the axial, sagittal, and coronal views. We then combined the results from the feature extraction of these three cross-sections. In this paper, we present a mobile-based classification system for Alzheimer's that offers greater flexibility for users.

2. METHOD

2.1. Clinical dementia rating

The clinical dementia rating (CDR) is a system used to assess dementia severity in individuals with Alzheimer’s disease. The CDR is divided into five levels: i) CDR-0 indicates no cognitive impairment, ii) CDR-0.5 represents very mild dementia, iii) CDR-1 signifies mild dementia, iv) CDR-2 corresponds to

moderate dementia, and v) CDR-3 reflects severe dementia. This system helps to quantify the degree of cognitive impairment in patients. In this experiment, we only use scale 0 to represent the condition of normal patients. Scales 1 and 2 are utilized solely to facilitate the classification of patients with Alzheimer's disease. In this experiment, we use scale 0 to represent the condition of normal patients. Scales 1 and 2 are used only to aid in the classification of patients with Alzheimer's disease. This approach is necessary because the mobile system we developed is designed for the early detection of Alzheimer's. If a patient is found to have a CDR score of 1 or 2, a referral will be made for further examination by a doctor.

2.2. Segmentation

The image segmentation process isolates the hippocampus from the background in an image. This process converts the image into a binary format, where the area representing the hippocampus appears white, and the background remains black. By performing this segmentation, we simplify the feature extraction process, reducing the number of features the SVM needs to analyze. In this research, we employ the threshold segmentation method called Double Thresholding, for which we manually determine the threshold values. Double Thresholding uses two thresholds to divide the image into three distinct regions. This approach allows for more precise object separation while minimizing the impact of noise [26]. We utilize Double Thresholding because single thresholding yields unsatisfactory results, with either too many pixels at a low threshold or too few at a high threshold. We use Python and several open-source libraries in this segmentation program experiment, including Scikit-image, OpenCV, and NumPy.

2.3. Feature extraction

After image segmentation, the next step is feature extraction, which aims to obtain characteristic information from the hippocampus. The features that distinguish the grayscale hippocampus include area, perimeter, and diameter. These features are determined by analyzing the decreased hippocampal area observed in the MRI images. Based on these features, images can be classified into three categories: CDR0, CDR1, and CDR2. The feature extraction stage produces a dataset organized into a spreadsheet file with the extension *.xlsx. This dataset will be input for machine learning modeling to perform classification tasks. In total, there are 42 data points available, consisting of 28 CDR0 images, 12 CDR1 images, and 2 CDR2 images. We have 10 CDR0 images, 8 CDR1 images, and 2 CDR2 images for testing purposes. Given the relatively small size of the dataset, we applied data augmentation to the training data to enhance accuracy. To perform feature extraction, we use the open-source library OpenCV in Python to simplify our coding process.

2.4. Implementation of the SVM model

One of the machine learning classification methods is SVM, which can be used to predict the data classes. SVM separates data into two or more classes based on the margin created by a hyperplane or decision boundary. A hyperplane is a line or surface that divides the data classes, while the margin is the shortest distance between the data points of a class and the hyperplane. In this research, we implemented SVM using the Python programming language and the Scikit-learn library. To utilize SVM, we employed the 'SVC' (Support Vector Classification) function, which allowed us to perform a three-class classification. This approach is consistent with our research objective of classifying the CDR level into three distinct classes. To achieve the best accuracy, SVM offers hyperparameters that can be adjusted to fit the specific data set. Unlike parameters derived during training, hyperparameters are set manually before training begins. Various hyperparameters are available, and we adjusted them to enhance the performance of the SVM model; i) **Kernel**: The kernel function defines the similarity between data points. Different kernels can capture various relationships among the data points, and the choice of kernel can significantly affect the performance of the SVM. Common types of kernels include linear, polynomial, radial basis functions (RBF), and sigmoid. ii) **C**: The C parameter in SVM is a hyperparameter that controls the trade-off between the margin and the amount of training error. A more considerable C value places a higher penalty on training errors, resulting in a smaller margin but potentially better generalization performance, iii) **Gamma**: The gamma parameter in SVM controls the influence of support vectors on the decision boundary. A more considerable gamma value means that nearby support vectors have a more substantial influence, while a smaller value indicates that distant support vectors exert less influence, and iv) **Degree**: In a polynomial kernel, the degree hyperparameter determines the degree of the polynomial used to transform the input data. This kernel interacts with the input data's dimensionality. In this context, hyperparameter settings can be adjusted automatically to enhance time efficiency. Automatic hyperparameter tuning can be performed using the "grid search" function in the Scikit-learn library. Grid search is a technique that helps find optimal hyperparameters for a machine learning model by exhaustively searching through a predefined range of hyperparameter values to identify the combination that yields the best performance.

3. RESULTS AND DISCUSSION

3.1. Segmentation and feature extraction

In the preprocessing stage of the mobile application we developed, we can select the area of the brain to be segmented. However, the small size of the hippocampus makes it challenging for users to cut the image accurately, resulting in uneven cuts or inaccurate targeting. We incorporate zooming and panning features in the image viewer to address this issue. Figure 1 displays a cropped section of the hippocampus from an MRI of the brain. The image used in the preprocessing cropping stage is a CDR2 brain image. The tiny hippocampus area makes it challenging for users to cut out the intended region, resulting in a cut area with a smaller array than masking. Consequently, an alternative segmentation method is employed alongside active contour, both serving the same purpose of separating the foreground from the background. The best results are achieved using binary thresholding.

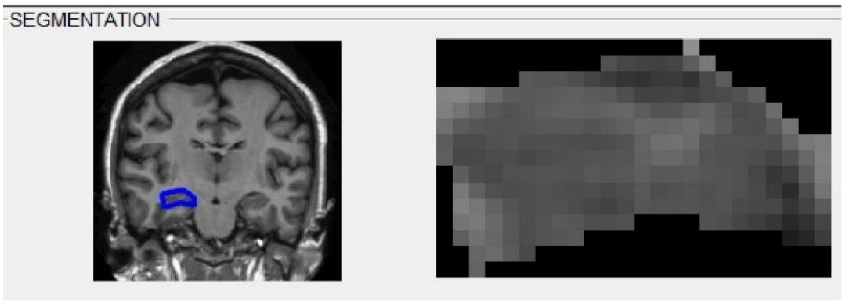

Figure 1. Cropping area of hippocampus

We utilize binary threshold segmentation because it effectively differentiates the area of interest from the background in an image. The threshold value is set based on the average pixel value of the image. This segmentation process ensures that the resulting image is informative and free from unnecessary details. This type of segmentation generates a binary image with only two possible values: 0 or 255. In this process, the object or area of interest will appear white, with a pixel value above the average. We compared several segmentation methods, including adaptive thresholding, Canny segmentation, in-range segmentation, binary inverse thresholding, and active contour. The results of this comparison are presented in Figure 2. As seen in the figure, these segmentation methods are ineffective in displaying the area objects we will utilize in this research. We used the hippocampus's area, perimeter, and diameter measurements for feature extraction. In the mobile application we developed, the area value is determined by counting the number of pixels with a value of 255. The diameter can be calculated using a specific formula that depends on the area value obtained. The perimeter is calculated by counting the number of pixels that form the edges.

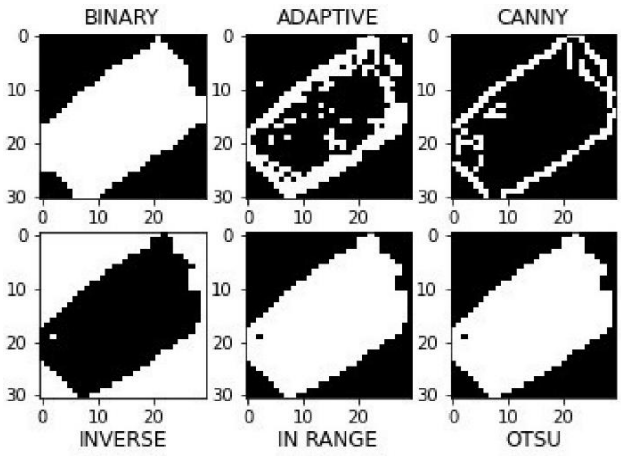

Figure 2. Results of comparison of several segmentation methods

3.2. Model evaluation

The mobile application we developed focuses on classifying the severity of Alzheimer's disease based on the CDR scale. The classification process aims to predict the CDR score from segmented images. This classification uses a trained model to determine the CDR level. The model is built using data obtained from feature extraction. To identify the CDR level, we utilize SVM algorithms, implemented through the scikit-learn library's GridSearchCV, which helps us find hyperparameters that maximize accuracy. The SVM also processes the feature extraction results in an Excel file. To make predictions on the test data, we use the function from the Scikit-learn library associated with the variable "grid_search" to generate predictions, specifically the method called "predict". We use the `train_test_split` function to divide the data into two sets: `X_train` and `X_test` for the features, and `y_train` and `y_test` for the target variables. The `test_size` parameter is set to 0.2, meaning that 20% of the data will be allocated for testing, while the remaining 80% will be used for training. We utilize the confusion matrix method to evaluate the classification model's performance. In this case, the confusion matrix is a 3x3 matrix that presents the values for True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN) across the three classification classes as described in Figure 3.

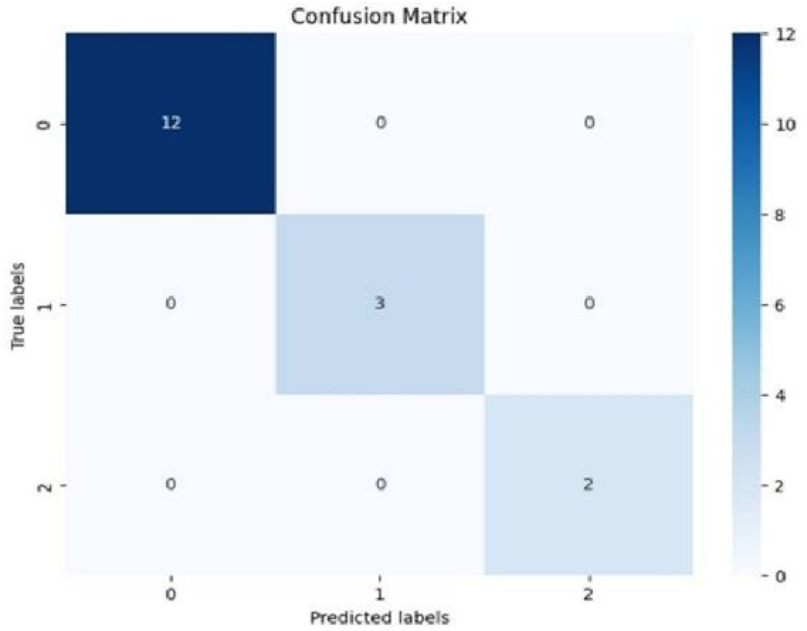

Figure 3. Confusion matrix for test data

The confusion matrix reveals the following results: true positive 1 (TP1) is 3, true positive 2 (TP2) is 2, and true negative (TN) is 12. This indicates that the model accurately classifies 12 instances of CDR0 test data, three cases of CDR1 test data, and two instances of CDR2 test data. Notably, there are no misclassified data points, as all false positive and false negative values (false positive 1/FP1, false positive 2/FP2, false positive 3/FP3, false negative 1/FN1, false negative 2/FN2, and false negative 3/FN3) are equal to 0. Consequently, based on the confusion matrix, the model's accuracy is 1, indicating perfect classification. Therefore, the model is suitable for application implementation. We utilize a classification report for model evaluation that presents a matrix containing accuracy, precision, recall, and F1-score. The values displayed reflect the performance of each target class within the model. This classification report aims to assess how well the model predicts each target class and evaluate the classification performance for each class. Precision measures how accurately the model classifies positive data, while recall assesses how effectively the model identifies all positive instances. The F1-score represents the harmonic mean of precision and recall. Generally, higher precision, recall, and F1-score values indicate better classification model performance. The classification report provides the evaluation results of the SVM model on the test data for the classes "CDR0," "CDR1," and "CDR2." The precision value for each class is 1.00, indicating that all predictions made by the model for these classes are accurate and none were misclassified. The recall value is also 1.00, which means the model successfully identifies all instances in each class. The F1-score, which combines

precision and recall to give a more comprehensive view of model performance, is recorded at 1.00. Lastly, the model accuracy is 1.00, indicating that it classifies all data correctly.

3.3. Android mobile application development

We developed an Android mobile application for hippocampus image classification using the Android development platform, Android Studio. The SVM machine learning model we created [24] It was saved using the "Pickle" library with a *.pkl file extension. This Python script will be integrated into the Android application using Chaquopy, which serves as an external plugin for Android Studio since the native languages supported by Android Studio are only Java and Kotlin. The application we created is HIPPA (Hippocampus Prediction for Pre-detection of Alzheimer's).

a) Front-End/user interface (UI) development

In Android application development, creating the front-end or UI is the first step before beginning the logical programming of the application. For this UI development, we designed six layouts using the XML programming language in Android Studio. These six layouts include MainActivity, coronal_image_insert, axial_image_insert, sagittal_image_insert, result_predict, and GuideActivity.

b) Back-end development

Back-end development ensures that the created UI functions properly and produces the desired output. In this phase, programming is carried out using the Kotlin programming language. Each activity layout is programmed in its own Kotlin activity or class. The corresponding layouts have six Kotlin activities: ActivityMain, GuideActivity, coronal_image_insert, axial_image_insert, sagital_image_insert, and result_predict. In this project, we utilize several external libraries to enable various functions within the application.

c) Implementing Python modules in android studio

This section will implement a Python SVM machine learning model in Android Studio using the Kotlin programming language. Android Studio natively supports Java and Kotlin, so we will use an external Chaquopy plugin to integrate Python code. Chaquopy enables Python programming within Android Studio by allowing us to create Python functions that perform computations and return results. The Kotlin program will then call these Python functions and store the returned values in Kotlin variables.

d) Implementing Python program

The Python program has been integrated into the Kotlin program by importing the necessary Python module and calling its primary function from Kotlin. To use values from the Python module within the Kotlin program, we need to store these values in Kotlin variables. To successfully call the Python module, it is essential to ensure that the Chaquopy module is included in the Kotlin program. Once we have retrieved all the expected return values from the Python function and stored them in Kotlin variables, we can access these variables and utilize the values as needed throughout the Kotlin program.

3.4. Application testing and evaluation

This stage involves conducting trials and evaluations of the Android applications that have been developed. This stage aims to understand how the application classifies images and identify its shortcomings. We evaluate the application by creating a classification table and a confusion matrix to assess its accuracy in classifying the CDR level of Alzheimer's disease. Figure 4 displays the main page of the HIPPA application we have developed. The application is designed in Indonesian because its primary users are Health Centers and Village Health Posts, where most users only understand the Indonesian language.

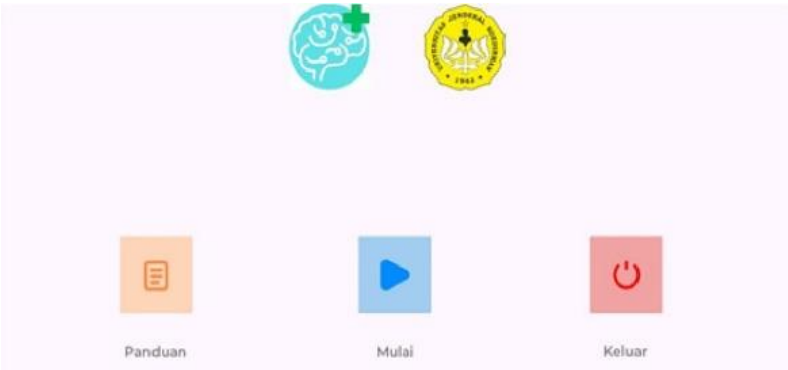

Figure 4. Application home page

This evaluation aims to assess the application's accuracy in performing CDR classification on the hippocampus. It includes two methods: creating a table of image classification results and generating a confusion matrix. We will evaluate the application using 20 test images, divided into 10 samples of CDR0, eight samples of CDR1, and two samples of CDR2. Table 1 shows the performance test results of the mobile application we developed.

Table 1. Test data classification results

No	Image	Original class	Prediction	Result
1	Axial1	CDR0	CDR0	True
2	Axial2	CDR0	CDR0	True
3	Axial3	CDR0	CDR0	True
4	Axial4	CDR0	CDR0	True
5	Sagital1	CDR0	CDR0	True
6	Aksial5	CDR0	CDR0	True
7	Sagital2	CDR0	CDR0	True
8	Axial6	CDR0	CDR0	True
9	**Sagital3**	**CDR0**	**CDR1**	**False**
10	Sagital4	CDR0	CDR0	True
11	Coronal1	CDR1	CDR1	True
12	Sagital5	CDR1	CDR1	True
13	Axial7	CDR1	CDR1	True
14	**Coronal2**	**CDR1**	**CDR2**	**False**
15	Coronal3	CDR1	CDR1	True
16	Coronal4	CDR1	CDR1	True
17	Axial8	CDR1	CDR1	True
18	Sagital6	CDR1	CDR1	True
19	Axial9	CDR2	CDR2	True
20	Sagital7	CDR2	CDR2	True

Table 1 presents the classification results for test data using the HIPPA application. The test data consists of 20 samples, including 10 from the CDR0 category, eight from the CDR1 category, and two from the CDR2 category. Table 1 shows that 9 of 10 CDR0 samples were classified correctly, while one was misclassified. For the CDR1 category, 7 out of 8 samples were correctly identified, with one misclassified. By analyzing the confusion matrix, we observe the following results: From 10 cases classified as CDR0, one was incorrectly classified as CDR1. Among 8 cases classified as CDR1, 1 case was misclassified as CDR2. All cases classified as CDR2 were accurately identified. In summary, for the CDR0 classification, the values are as follows: true positives (TP1) = 9, false positives (FP1) = 1, and false negatives (FN1) = 0. For the CDR1 classification, we have: true positives (TP2) = 7, false positives (FP2) = 0, and false negatives (FN2) = 1. Lastly, for the CDR2 classification, the values are: true positives (TP3) = 2, false positives (FP3) = 0, and false negatives (FN3) = 0. The accuracy of the classification results in the HIPAA application is 90%. This level of accuracy indicates that the application is quite effective at classifying images. Given the purpose of this application, this 90% accuracy should primarily be used for initial examinations or pre-detection and prevention.

4. CONCLUSION

The "Segmentation" module utilizes a double-thresholding method in the mobile application we developed. This approach effectively applies masks to areas with specific intensity levels, as demonstrated by the results from images processed through segmentation. Additionally, the machine learning model built with the SVM architecture performs exceptionally well in research, evident from the accuracy achieved during the training and testing phases.

Furthermore, the external Chaquopy plug-in used in this experiment seamlessly integrates Python into the Android Studio environment. The application's success in segmenting images, returning feature values, and classifying the identified features is evidenced by this. After thoroughly evaluating the HIPPA application, we concluded that it operates effectively and accurately pre-detects levels of CDR with an impressive accuracy rate of 90%.

FUNDING INFORMATION

This research was funded by LPDP (*Lembaga Pengelola Dana Pendidikan*) through the National Research and Innovation Agency or *Badan Riset dan Inovasi Nasional* (BRIN) through the "*Riset Inovasi*

untuk Indonesia Maju Scheme 2024" with contract number 39/IV/KS/05/2023 and 3080/UN23/PT.01.02/2024.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Retno Supriyanti	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Muhammad Putra Yubiksana		✓		✓				✓						
Bintang Abelian Mahardika Wijonarko		✓		✓				✓						
Yogi Ramadhani					✓		✓							
Muhammad Syaiful Aliim					✓		✓							
Mohammad Irham Akbar			✓								✓			
Haris Budi Widodo				✓						✓				
Wahyu Widanarto							✓			✓				
Muhammad Alqaaf					✓							✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

ETHICAL APPROVAL

The research related to human use has been complied with all the relevant national regulations and institutional policies in accordance with the tenets of the Helsinki Declaration and has been approved by the authors' institutional review board or equivalent committee; or: The research related to animal use has been complied with all the relevant national regulations and institutional policies for the care and use of animals.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] G. McKhann, D. Drachman, M. Folstein, R. Katzman, D. Price, and E. Stadlan, "Clinical diagnosis of Alzheimer's disease: Report of the NINCDS- ADRDA Work Group* under the auspices of Department of Health and Human Services Task Force on Alzheimer's Disease," *Neurology*, vol. 34, no. 7, pp. 939–939, 1984.

[2] D. Georges, O. Olle, J. Zoobo, B. Ghislain, and A. Alo, "Application and comparison of K - means and PCA based segmentation models for Alzheimer disease detection using MRI," *Discover Artificial Intelligence*, vol. 4, no. 11, 2024, doi: 10.1007/s44163-024-00106-7.

[3] O. A. Dara, J. M. Lopez-Guede, H. I. Raheem, J. Rahebi, E. Zulueta, and U. Fernandez-Gamiz, "Alzheimer's Disease Diagnosis Using Machine Learning: A Survey," *Applied Sciences (Switzerland)*, vol. 13, no. 14, 2023, doi: 10.3390/app13148298.

[4] S. Chaudhary, R. Nehra, K. Kumar, S. Dahiya, D. Nandan, and S. Kumar, "Dementia Alzheimer's Disease Diagnosis Using Convolutional Neural Networks on Two-Dimensional MRI Slices," *Revue d'Intelligence Artificielle*, vol. 38, no. 1, pp. 81–91, 2024, doi: 10.18280/ria.380109.
[5] T. Prasath and V. Sumathi, "Identification of Alzheimer's Disease by imaging: a comprehensive review," *International Journal of Environmental Research and Public Health*, vol. 20, no. 2, 2023, doi: 10.3390/ijerph20021273.
[6] E. H. Ali, S. Sadek, G. Z. El Nashef, and Z. F. Makki, "Advanced Integration of Machine Learning Techniques for Accurate Segmentation and Detection of Alzheimer's Disease," *Algorithms*, vol. 17, no. 5, 2024, doi: 10.3390/a17050207.
[7] P. Gayathri *et al.*, "Deep Learning Augmented with SMOTE for Timely Alzheimer's Disease Detection in MRI Images," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 2, pp. 499–508, 2024, doi: 10.14569/IJACSA.2024.0150253.
[8] S. S. Bamber and T. Vishvakarma, "Medical image classification for Alzheimer's using a deep learning approach," *Journal of Engineering and Applied Science*, vol. 70, no. 1, pp. 1–19, 2023, doi: 10.1186/s44147-023-00211-x.
[9] S. S. Abd-Aljabar, N. M. Basheer, and O. I. Alsaif, "Alzheimer's diseases classification using YOLOv2 object detection technique," *International Journal of Reconfigurable and Embedded Systems*, vol. 11, no. 3, pp. 240–248, 2022, doi: 10.11591/ijres.v11.i3.pp240-248.
[10] C. Mahanty *et al.*, "Effective Alzheimer's disease detection using enhanced Xception blending with snapshot ensemble," *Scientific Reports*, vol. 14, no. 1, pp. 1–18, 2024, doi: 10.1038/s41598-024-80548-2.
[11] J. Wang, S. Wen, W. Liu, X. Meng, and Z. Jiao, "Deep joint learning diagnosis of Alzheimer's disease based on multimodal feature fusion," *BioData Mining*, vol. 17, no. 1, 2024, doi: 10.1186/s13040-024-00395-9.
[12] C. L. Saratxaga *et al.*, "Mri deep learning-based solution for alzheimer's disease prediction," *Journal of Personalized Medicine*, vol. 11, no. 9, 2021, doi: 10.3390/jpm11090902.
[13] S. Jahan *et al.*, "Explainable AI-based Alzheimer's prediction and management using multimodal data," *PLoS ONE*, vol. 18, no. 11 November, pp. 1–27, 2023, doi: 10.1371/journal.pone.0294253.
[14] N. K. Parvatham and L. P. Maguluri, "Improved Decision Support System for Alzheimer's Diagnosis Using a Hybrid Machine Learning Approach with Structural MRI Brain Scans," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 8, pp. 476–484, 2024, doi: 10.14569/IJACSA.2024.0150847.
[15] A. D. Arya *et al.*, "A systematic review on machine learning and deep learning techniques in the effective diagnosis of Alzheimer's disease," *Brain Informatics*, vol. 10, no. 1, Jul. 2023, doi: 10.1186/s40708-023-00195-7.
[16] N. Pasnoori, T. Flores-Garcia, and B. D. Barkana, "Histogram-based features track Alzheimer's progression in brain MRI," *Scientific Reports*, vol. 14, no. 1, pp. 1–13, 2024, doi: 10.1038/s41598-023-50631-1.
[17] X. Tian *et al.*, "Deep Residual-in-Residual Model-Based PET Image Super-Resolution with Motion Blur," *Electronics (Switzerland)*, vol. 13, no. 13, 2024, doi: 10.3390/electronics13132582.
[18] B. Şener, K. Acici, and E. Sümer, "Categorization of Alzheimer's disease stages using deep learning approaches with McNemar's test," *PeerJ Computer Science*, vol. 10, pp. 1–31, 2024, doi: 10.7717/peerj-cs.1877.
[19] A. P. Begum and P. Selvaraj, "Traitement du Signal NOVE-Seg : An Effective Framework for Detection of Alzheimer Disease Using Opti-FRCNN on Brain MRI," vol. 41, no. 5, pp. 2729–2741, 2024.
[20] X. Li, B. Gong, X. Chen, H. Li, and G. Yuan, "Alzheimer's disease image classification based on enhanced residual attention network," *PLoS ONE*, vol. 20, no. 1 January, pp. 1–15, 2025, doi: 10.1371/journal.pone.0317376.
[21] R. Ghadami and J. Rahebi, "Alzheimer's Prediction Methods with Harris Hawks Optimization (HHO) and Deep Learning-Based Approach Using an MLP-LSTM Hybrid Network," *Diagnostics*, vol. 15, no. 3, 2025, doi: 10.3390/diagnostics15030377.
[22] R. Supriyanti, Y. Ramadhani, and E. Wahyudi, "Hippocampus's volume calculation on coronal slice's for strengthening the diagnosis of Alzheimer's," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 21, no. 1, pp. 123–132, 2023, doi: 10.12928/TELKOMNIKA.v21i1.20746.
[23] R. Supriyanti, A. R. Subhi, E. J. Ashari, F. Ahmad, Y. Ramadhani, and H. B. Widodo, "Simple Classification of the Alzheimer's Severity in Supporting Strengthening the Diagnosis of Patients based on ROC Diagram," in *IOP Conference Series: Materials Science and Engineering*, Dec. 2020, vol. 982, no. 1, doi: 10.1088/1757-899X/982/1/012007.
[24] R. Supriyanti *et al.*, "Support vector machine method for classifying severity of Alzheimer's based on hippocampus object using magnetic resonance imaging modalities," *International Journal of Electrical and Computer Engineering*, vol. 14, no. 6, pp. 6322–6331, 2024, doi: 10.11591/ijece.v14i6.pp6322-6331.
[25] R. Supriyanti, A. R. Subhi, Y. Ramadhani, and H. B. Widodo, "Coronal Slices Segmentation of MRI Images using Active," *Journal of Engineering Science and Technology*, vol. 14, no. 3, pp. 1672–1686, 2019.
[26] Rafael Gonzalez and Richard Woods, *Digital image processing*, 4th ed., vol. 1. New York: Pearson, 2017.

BIOGRAPHIES OF AUTHORS

Retno Supriyanti is a professor at Electrical Engineering Department, Jenderal Soedirman University, Indonesia. She received her PhD in March 2010 from Nara Institute of Science and Technology, Japan. She also received her M.S. and bachelor's degrees in 2001 and 1998, respectively, from the Electrical Engineering Department, Gadjah Mada University, Indonesia. Her research interests include image processing, computer vision, pattern recognition, biomedical applications, e-health, telehealth, and telemedicine. She can be contacted at email: retno_supriyanti@unsoed.ac.id.

Muhammad Putra Yubiksana received his bachelor's degree from the Electrical Engineering Department, Jenderal Soedirman University Indonesia. Her research interest is in the field of image processing. He can be contacted at email: putra.yubiksan012@gmail.com.

Bintang Abelian Mahardika Wijonarko received his bachelor's degree from the Electrical Engineering Department, Jenderal Soedirman University Indonesia. Her research interest is in the field of image processing. He can be contacted at email: bintang.wijonarko@mhs.unsoed.ac.id.

Yogi Ramadhani is an academic staff at Electrical Engineering Department, Jenderal Soedirman University, Indonesia. He received his MS Gadjah Mada University Indonesia, and his Bachelor degree from Jenderal Soedirman University Indonesia. His research interest including computer network, decision support system, telemedicine and medical imaging. He can be contacted at email: yogi.ramadhani@unsoed.ac.id.

Muhammad Syaiful Aliim is a lecturer at the Electrical Engineering Department, Jenderal Soedirman University, Indonesia. He received his Master degree in Electrical Engineering from Indonesia University, Indonesia, from 2014 until 2016, and his bachelor's degree in Electrical Engineering from Jenderal Soedirman University, Indonesia, from 2008 until 2012. Before working as a lecturer, he worked as a Java Programmer at Ace Global Consulting and Integration (IT Consulting Company focused on the Retail Industry) from 2013 until 2019. His research interests include machine learning, deep learning, computer vision, machine learning on augmented reality, and the internet of things. He can be contacted at this email: muhammad.syaiful.aliim@unsoed.ac.id.

Mohammad Irham Akbar is an academic staff at Electrical Engineering Department, Jenderal Soedirman University, Indonesia. He received his MS Gadjah Mada University Indonesia, and his Bachelor degree from Jenderal Soedirman University Indonesia. His research interest including computer network, decision support system, data mining and medical imaging. He can be contacted at email: mohammad.irham@unsoed.ac.id.

Haris Budi Widodo is an academic staff at Public Health Department, Jenderal Soedirman University, Indonesia. He received his Ph.D. from Airlangga University Indonesia. Also, He received his M.S degree and bachelor degree from Gadjah Mada University Indonesia. His research interest including public health, e-health and telemedicine. He can be contacted at email: haris.bwidodo@unsoed.ac.id.

Wahyu Widanarto completed his undergraduate degree majoring in Physics at Brawijaya University, his master's degree majoring in Physics at the Bandung Institute of Technology, and his Ph.D. at the Universitaet der Bundeswehr Munich Germany in the field of Institut fuer Physik, Mikrosystemtechnik. His current research is on the development of carbon materials for sensors. He can be contacted at email: haris.bwidodo@unsoed.ac.id.

Muhammad Alqaaf received his bachelor degree from Department of Electrical Engineering, Jenderal Soedirman University, Indonesia; M.S. degree in Nara Institute of Science and Technology, Japan. Currenly he is a doctor student in Nara Institute of Science and Technology, Japan. His research interest image processing field and bioinformatics. He can be contacted at email: muhammad.alqaaf_subandoko.mb5@is.naist.jp.