##### AUTOMATED DETECTION OF ALZHEIMER DISEASE USING CNN ALGORITHM

##### A PROJECT REPORT

###### ***Submitted by***

|  |  |
| --- | --- |
| DINESH KUMAR V | [211419104070] |
| DIWAKAR V | [211419104073] |
| DEVENTHIRAN R | [211419104055] |

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

IN

# COMPUTER SCIENCE AND ENGINEERING



PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

##### APRIL 2023

**BONAFIDE CERTIFICATE**

Certified that this project report **“AUTOMATED DETECTION OF ALZHEIMER DISEASE USING CNN ALGORITHM”** is the bonafide work of **“DINESH KUMAR V (211419104070), DIWAKAR V(211419104073), DEVENTHIRAN R (211419104055)”** who carried out the project work under my supervision.

**SIGNATURE SIGNATURE**

**Dr.L.JABASHEELA,M.E.,Ph.D., Dr.M.SANGEETHA,M.TECH.,Ph.D.,**

**HEAD OF THE DEPARTMENT SUPERVISOR**

**ASSOCIATE PROFESSOR**

DEPARTMENT OF CSE, DEPARTMENT OF CSE,

PANIMALAR ENGINEERING COLLEGE, PANIMALAR ENGINEERING COLLEGE,

NASARATHPETTAI, NASARATHPETTAI,

POONAMALLEE, POONAMALLEE,

CHENNAI-600 123. CHENNAI-600 123.

Certified that the above candidate(s) was/ were examined in the End Semester Project Viva-Voce Examination held on...........................

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**DECLARATION BY THE STUDENT**

We **DINESH KUMAR V (211419104070) ,DIWAKAR V (211419104073) ,DEVENTHIRAN R (211419104055)** hereby declare that this project report titled **“AUTOMATED DETECTION OF ALZHEIMER DISEASE USING CNN ALGORITHM”,** under the guidance of **Dr.M.SANGEETHA M.Tech.,Ph.D.** is the original work done by us and we have not plagiarized or submitted to any other degree in any university by us.

DINESH KUMAR V

DIWAKAR V

DEVENTHIRAN R

**ACKNOWLEDGEMENT**

We would like to express our deep gratitude to our respected Secretary and Correspondent **Dr.P.CHINNADURAI, M.A., Ph.D.** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We express our sincere thanks to our beloved Directors **Tmt.C.VIJAYARAJESWARI**, **Dr.C.SAKTHI KUMAR, M.E.,Ph.D** and **Dr.SARANYASREE SAKTHI KUMAR B.E.,M.B.A.,Ph.D.,** for providing us with the necessary facilities to undertake this project.

We also express our gratitude to our Principal **Dr.K.MANI, M.E., Ph.D.** who facilitated us in completing the project.

We thank the Head of the CSE Department, **Dr. L.JABASHEELA, M.E.,Ph.D.,** for the support extended throughout the project.

We would like to thank our parents, friends, project Guide **Dr.M.SANGEETHA M.Tech.,Ph.D.,** and coordinator **Dr.N.PUGHAZENDI M.E., Ph.D.,** and all the faculty members of the Department of CSE for their advice and encouragement for the successful completion of the project.

DINESH KUMAR V

DIWAKAR V

DEVENTHIRAN R

**ABSTRACT**

Many different architectures of Convolutional Neural Networks (CNNs) have been developed for use in classifying images and recognizing objects. When it comes to image-based categorization, handling hundreds of MRI image slices that are essentially identical across patients is a challenging task for CNN. By utilizing a 2D CNN design, it becomes difficult to confidently categorize a large number of individuals as having Alzheimer's disease, mild cognitive impairment, or normal cognition. To solve this problem, we have streamlined the concept of patient classification based on 3D MRI while still giving due credit to the 2D features derived from the CNN framework. Here, we share our approach to extracting 2D features from MRI scans in a format that can be used in a classification system. Our experiment demonstrates the outcome of categorizing 3 patient participants into 2 groups. After reducing the dimensionality of a 2D image with principal component analysis and truncated sparse encoding (PCA+TSNE), we used a convolutional neural network (CNN) to extract generic features for classification. Despite the lackluster performance, this seems to be an improvement over probability-based categorization using a CNN that was trained from scratch. The created feature is highly malleable and can be fine-tuned to improve precision, responsiveness, and specificity.

TABLE OF CONTENTS

| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
| --- | --- | --- |
|  | **ABSTRACT** | v |
|  | **LIST OF TABLES** | ix |
|  | **LIST OF FIGURES** | x |
|  | **LIST OF SYMBOLS, ABBREVIATIONS** | xi |
| **1.** | **INTRODUCTION** | 1 |
|  | 1.1 Overview | 2 |
|  | 1.2 Problem Definition | 3 |
| **2.** | **LITERATURE SURVEY** | 4 |
| **3.** | **SYSTEM ANALYSIS** | 11 |
|  | 3.1 Existing System | 12 |
|  | 3.2 Disadvantages of Existing System | 12 |
|  | 3.3 Proposed system | 12 |
|  | 3.4 Feasibility Study | 12 |
|  | 3.5 Hardware Requirements | 13 |
|  | 3.6 Software Requirements | 13 |
|  | 3.7 Technologies Used | 13 |
|  | 3.7.1 Python | 14 |
|  | 3.7.2 Anaconda | 14 |
| **4.** | **SYSTEM DESIGN** | 17 |
|  | 4.1 Data Flow Diagram (DFD) | 17 |
|  | 4.2 UML Diagrams | 19 |
|  | 4.2.1 Use Case Diagram | 19 |
|  | 4.2.2 Class Diagram | 21 |
|  | 4.2.3 Sequence Diagram | 22 |
|  | 4.2.4 Activity Diagram | 23 |
|  | 4.2.5 Component Diagram | 24 |
| **5.** | **SYSTEM ARCHITECTURE** | 25 |
|  | 5.1 Architecture Diagram | 26 |
|  | 5.2 Algorithms | 27 |
|  | 5.2.1 Convolution Neural Network | 27 |
| **6.** | **SYSTEM IMPLEMENTATION** | 28 |
|  | 6.1 MODULE DESIGN SPECIFICATION | 29 |
|  | 6.1.1 Data collection and acquisition | 29 |
|  | 6.1.2 Data pre-processing | 29 |
|  | 6.1.3 Feature extraction | 30 |
|  | 6.1.3 Classsification Model | 30 |
|  | 6.1.4 Model evaluation | 30 |
|  | 6.1.5 Model deployment | 30 |
| **7.** | **SYSTEM TESTING** | 31 |
|  | 7.1 Software Testing | 32 |
|  | 7.1.1 Unit Testing | 32 |
|  | 7.1.2 Functional Testing | 32 |
|  | 7.1.3 Integration Testing | 33 |
|  | 7.1.4 White Box Testing | 33 |
|  | 7.1.5 Black Box Testing | 33 |
|  | 7.1.6 System Testing | 34 |
|  | 7.1.7 Output Testing | 34 |
|  | 7.1.8 User Acceptance Testing | 34 |
|  | 7.2 Test Cases | 35 |
| **8.** | **CONCLUSION & FUTURE ENHANCEMENT** | 37 |
|  | 8.1 Conclusion | 38 |
|  | 8.2 Future Enhancements | 38 |
|  | **APPENDICES** | 39 |
|  | A.1Coding | 40 |
|  | A.2 Sample Screens | 48 |
|  | **REFERENCES** | 55 |
|  |  |  |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **TABLE NO.** | **TABLE DESCRIPTION** | **PAGE NO.** |
|  |  |  |
| 7.2.1 | Test Case For Image Upload | 29 |
| 7.2.2 | Test Case For Identify and Detect | 30 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO.** | **FIGURE DESCRIPTION** | **PAGE NO.** |
| 4.1.1 | Level 0 of Data flow diagram | 14 |
| 4.1.2 | Level 1 of Data flow diagram | 15 |
| 4.1.3 | Level 2 of Data flow diagram | 15 |
| 4.2.1 | Use Case Diagram | 17 |
| 4.2.2 | Class Diagram | 18 |
| 4.2.3 | Sequence Diagram | 19 |
| 4.2.4 | Activity Diagram | 20 |
| 4.2.5 | Component Diagram | 21 |
| 5.1 | Architecture Diagram | 22 |
| 5.2.1.1 | Classify Time Series | 23 |
| A.2.1 | Anaconda Prompt Console | 40 |
| A.2.2 | UI of the Jupyter Notebook | 41 |
| A.2.3 | Training Folder | 41 |
| A.2.4 | Testing Folder | 42 |
| A.2.5 | Main Source File | 42 |
| A.2.6 | Epoches | 43 |
| A.2.7 | Epoches | 43 |
| A.2.8 | Validation Graph(Model Loss) | 44 |
| A.2.9 | Validation Graph(Model Accuracy) | 45 |
| A.2.10 | Final Input and output | 46 |

## LIST OF SYMBOLS, ABBREVIATIONS

|  |  |
| --- | --- |
| **AD** | Alzheimer’s Disease |
| **ADNI** | Alzheimer’s Disease Neuroimaging Initiative |
| **BAC** | Balanced Accuracy |
| **CDR** | Clinical Dementia Rating |
| **CI** | Confidence Interval |
| **CSF** | Cerebrospinal Fluid |
| **CV** | Cross-Validation |
| **DT** | Decision Tree |
| **ET** | Extremely Randomized Tree |
| **FN** | False Negatives |
| **LDA** | Linear Discriminant Analysis |
| **LR** | Logistic Regression |