**ABSTRACT**

This project “**BHARATHI**” is aimed at developing software utility which will recognise handwritten characters of Tamil language script and can be accessed through an Input Method Editor.

With rising touch enabled smart-phones and tablet market in India, there is strong need to develop software that provides native language support. This utility enables native users to overcome language barrier in access to technology, by recognizing Tamil characters and numerals. It is also helpful in recognizing special symbols. It engulfs the concept of neural network.

One of the primary means by which computers are endowed with human-like abilities is through the use of a neural network. Neural networks are particularly useful for solving problems that cannot be expressed as a series of steps, such as recognizing patterns, classifying them into groups, series prediction and data mining. The neural network which is trained for classification is designed to take input samples of a hand written data pattern, which then attempts to classify them into groups to determine if the input data matches a pattern that it has memorized.

This project is targeted on Android based Aakash tablets and concerns detecting free handwritten characters through touch gestures. It can be further developed to recognize the characters of different languages.

**TABLE OF CONTENTS**

**CHAPTER NO. TITLE PAGE NO.**

**ABSTRACT iii**

**LIST OF TABLES viii**

**LIST OF FIGURES ix**

**LIST OF ABBREVIATIONS x**

**1 INTRODUCTION 1**

1.1 Aim 1

1.2 Overview of the project 1

1.3 Artificial Neural Network 1

1.3.1 Kohonen Neural Networks 2

1.3.2 How KNN Learns 3

**2 LITERATURE SURVEY 5**

2.1Existing System 5

2.2 On Screen Input Methods 5

2.3 Drawbacks 8

2.4 Proposed System 9

**3 REQUIREMENT SPECIFICATION 10**

3.1 Introduction 10

3.1.1 Purpose 10

**CHAPTER NO. TITLE PAGE NO.**

3.1.2 Scope 10

3.2 General Description 10

3.2.1 Product Perspective 10

3.2.2 Product functions overview 10

3.2.3 User Characteristics 11

3.2.4 Operating Environment 12

3.2.5 Design & Implementation Constraint 12

3.2.6 Assumptions and Dependencies 12

3.3 Feasibility analysis 13

3.3.1 Economic Feasibility 13

3.3.2 Technical Feasibility 13

3.3.3 Social Feasibility 13

3.4 System Features 13

3.5 External Interface Requirements 14

3.5.1 User Interface 14

3.5.2 Hardware Interface 14

3.5.3 Software Interface 14

3.6 Nonfunctional Requirements 14

3.6.1 Performance Requirements 14

**CHAPTER NO. TITLE PAGE NO.**

3.6.2 Safety requirements 14

3.6.3 Software Quality Attributes 15

**4 DETAILED DESIGN 16**

4.1 Introduction 16

4.2 Lifecycle of IME 16

4.3 Architecture of Bharathi 17

4.4 Data Flow Design1 19

4.5 UML Diagrams 20

4.5.1 Use Case Diagram 20

4.5.2 Class Diagram 21

4.5.3 Activity Diagram 22

4.5.4 Sequence Diagram 24

4.5.5 Component Diagram 24

**5 MODULES 25**

5.1 Training module 25

5.2 Recognize module 26

**6 CONCLUSION & FUTURE ENHANCEMENT 29**

6.1 Conclusion 29

6.2 Future enhancement29

**CHAPTER NO. TITLE PAGE NO.**

**APPENDIX A DEVELOPMENT TOOLS 30**

**APPENDIX B RUNTIME ENVIRONMENT 32**

**APPENDIX B SAMPLE CODE 35**

**APPENDIX D SAMPLE OUTPUT SCREENSHOTS 46**

**REFERNCE S 49**

**LIST OF TABLES**

**TABLE NO. TITLE PAGE NO.**

3.1  Hardware Requirements 12

3.2 Software Requirements 12

**LIST OF FIGURES**

**FIGURE NO. TITLE PAGE NO.**

1.1 Self Organizing Maps Demo 2

1.2 Training the KNN 3

2.1 Sample Android Home Screen 5

2.2 Accessing Soft Keyboard 6

2.3 Full Sized Keyboard 7

2.4 Tamil Soft Keyboard 8

4.1 The Lifecycle Diagram of IME 16

4.2 Architecture Diagram of Bharathi 17

4.3 Dataflow Diagrams For Engine 19

4.4 Use Case Diagram for IME 20

4.5 Class Diagram for Bharathi 21

4.6 Recognize activity diagram for Engine 22

4.7 Training activity diagram for Engine 23

D.1 Setting Up Bharathi 46

D.2 IME Screenshots 48

**LIST OF ABBREVIATIONS**

ADT Android Developer Tools

SDK Software Development Kit

DVM Dalvik Virtual Machine

GUI Graphical User Interface

HWR Hand Writing Recognition

ANN Artificial Neural Networks

SOM Self Organizing Maps

I/O Input and Output

PE Processing Elements

KNN Kohonen Neural Networks

UML Unified Modelling Language

IMF Input Method Framework