# OPTICAL CHARACTER RECOGNITION ON THE ANDROID OPERATING SYSTEM FOR KANNADA CHARACTERS USING KOHONEN NEURAL NETWORK

**Chandrashekhar M V[1], Santosh Kumar M[2], Nasreen Taj M.B [3], Asha K [4]**

[1, 2 ,3, 4]*Assistant Professor, Dept. of ISE, G.M. Institute of Technology, Davangere , Karnataka , (India)*

**ABSTRACT**

*This paper focuses on an application that performs Kannada OCR (Optical Character Recognition) in hand held devices. The objective is to make use of the visual capabilities of the built in camera of Android devices to extract text from Kannada sign boards, newspapers etc., and use the same platform to implement the OCR technology with the help of Kohonen algorithm. The image is processed, eliminating small distortions present. The image is then converted to a grey-scale image, which is segmented and the result is displayed along with a vocal output.*

***Keywords* - *Optical Character Recognition, JAVA, JVM, Android***

## I. INTRODUCTION

This paper will enable users to use the application developed for regular purposes. This will enable them to read and understand the various boards, advertisements, shop names, addresses, business cards etc. Extracting text from these require accurate recognition of the characters amidst different environmental conditions like luminosity, rotation, reflection, scaling among others. Since creating an algorithm with a one hundred percent correct recognition rate is almost impossible due to noise and different font styles, but one can design character recognition algorithms and develop applications with these failures in mind so that when mistakes are inevitably made they will at least be understandable and predictable to the person using them.

Recognizing characters present in an image makes the processing of various different kinds of data comparatively easier. The computing device can be outfitted with a camera so that software in the device can use this to take pictures of the data available like a hand written text and give the characters written as an output. Proper user interface has to be created which should help the user to enable an easier way to use this software.

## II. LITERATURE SURVEY

In day-to-day life there are a lot of situations where one is unable to understand scripts in other languages. An automated system for reading documents or cards has also gained importance in various fields. All of these can be addressed by optical character recognition software. The project will be useful mainly when it is extended to different regional languages. Android gives a platform where one can easily install applications and use them effortlessly. Most of the Android supporting devices also contain a built in camera, which are capable of taking good quality pictures. This project aims to make use of the Android platform [1] to develop

an application for recognizing characters by taking pictures of them through built in camera with a relatively dependable efficiency.

OCR has been in development for almost 80 years, as patent for an OCR machine was filed by a German named Gustav Tauschek in 1929, and an American patent was filed subsequently 1935. OCR has many applications, including use in the postal service, language translation and digital libraries. Little or no development was done between the years of 1996 and 2006.

Early optical character recognition could be traced to activity around two issues: expanding telegraphy and creating reading devices for the blind. Later it was continued to develop OCR technology for data entry. It was proposed to be used in photographing data records and then, using photocells, matching the photos against a template containing the desired identification pattern.

An OCR engine was developed by Hewlett- Packard between 1985 and 1994. It is one of the most important applications of the OCR technology. It is most suitable for backend working. Apart from character recognition the software can also detect whether the text is mono spaced or proportional. Various papers have been presented on the OCR over the years. Some of them are:

The use of OCR for logo matching. The paper [2] gives an insight into logo matching where translation, scale and rotation of the image containing the images. The image is prepared by processing the image using various transformations. As the paper is dealing with logos and not characters of similar fonts and sizes they have used feature extraction for processing the image and for character retrieval. Various experiments like baseline technique, evaluation metrics are used to compare the accuracy of the application.

Paper [3] describes an accurate OCR for English. The paper mainly concentrates on business cards with fixed font and colour characters. The approach taken is a very simple one, comparing the characters with the one present in the database as English has only 26 alphabets. There is no use of any type of neural network like artificial or Kohonen neural network. The author uses a very soft approach but tries different experiments to prove that in OCR 100% accuracy is possible. This paper gives a very basic idea of the technology and introduces it to the beginners.

Paper [4] describes the application of OCR in scanning books. The main aim is to make the technology useful for reading e-texts and e-books. The unique words in vocabulary of the book are lined up against the outputs of the OCR. This is done repetitively till the number of such words become very less. Distance based alignment algorithm is used for alignment of the text. This is used for character recognition of books written in Spanish, French, English and German.

Paper [5] explains the optical character recognition using neural network for Bangla characters. The paper gives an object oriented modeling framework of a Kohonen based character recognition system. The paper provides an insight into the regional language, the challenges faced and the feature extraction method, which is used for the character detection. The paper helps to learn the implementation of OCR to Indian regional languages, as the number of characters including vowels, consonants and complicated letters are very much similar to most of the other Indian languages.

## III ALGORITHM USED

**3.1 Kohonen Neural Network:** As it is an unsupervised network i.e. only the input of the network is initialized and the output of the network is not set. In Kohonen neural network one of the neurons decides the output as it becomes the winning neuron. The weights associated with the input nodes are set between 1 and -1.

The input is normalized by using the vector length of the input data. For each neuron the output is calculated and the best neuron is selected as the winner. Thus every time a winner is selected and the network is adjusted using the weights such that the possibility of the winning neuron increases for the same input in future. The application being developed makes use of the Kohonen neural network and then for each character the network provides a winning neuron which helps to recognize the characters.

Steps involved in this algorithm:

1.Randomize the map's nodes' weight vectors

2.Traverse each input vector in the input data set

3.Traverse each node in the map

4.Use the Euclidean distance formula to find the similarity between the input vector and the map's node's weight vector

5.Track the node that produces the smallest distance (this node is the best matching unit, BMU)

6.Update the nodes in the neighborhood of the BMU (including the BMU itself) by pulling them closer to the input vector

7.Increase the count of iteration and repeat the process.

## IV. SYSTEM ARCHITECTURE

General Constraints:

1.End-user environment: The built in camera should be able to take good quality images so that the processing of it does not lead to ambiguous or undesired result.

2. Standards compliance: The API's used to develop the application should comply with the standard general API's of the Android version 2.3.

3. Interoperability requirements: Any device that will use or improve or integrate the application must comply with the standards, operating system and library requirements.

4. Memory and other capacity limitations: Very limited memory is needed. RAM should be of enough to run the application and also to process the image taken by mapping with the back end letters.

Development Methods:The optical character recognition is being done using the Kohonen [6] neural network. A preliminary design of the network is done with random number of input and number of output nodes. In the network an input is given but the output expected is not mentioned. With every character recognized the probability of detection of the characters increases. Different packages are used for implementing different activities like camera, result, voice output etc. The processing of image is designed and improved iteratively to reach a state from where the recognition can be done relatively easily. Unit testing is carried at each level after the development of each module and integration testing is carried out to localize the errors and to debug them. Various features of Android like logcat is used extensively to localize the errors.

## V. IMPLEMENTATION DETAILS

The working of the application is as follows:

1.The Kohonen neural network is set with random number of input and output nodes, with the inputs being known and the outputs unknown.

2. In Kohonen network one of the output nodes is each time selected as a winning output thus the whole

single process of detecting each character is decided with the help of an output node.

3.With each character recognized the network is adjusted so that the accuracy of the application increases for next character recognition.

4.The image taken is further processed by initially thinning the characters in the image.

5. The image is processed for noise removal.

6. Thus eliminating the little distortions present.

7. Every pixel of the image is analyzed and if the value of any pixel is greater than a threshold value the pixel is made completely black or else made completely white. Thus the characters are made highly contrasting when compared with the background.

8.The image is segmented to separate and distinguish the characters from one another and processed individually to recognize each of them. The result is displayed on the result activity along with the vocal output.

## VI. CRITICLE INFORMATION

There were a number of challenges faced while implementing the OCR. Few among as mentioned below.

**Background shadows in the image:**While taking the image the background image may contain some shadows. These irrelevant shadows may cause a large distortion when the image is processed and these shadows become more prominent resulting in the wrong identification of characters. As processing the image results in removing the little distractions it also causes the shadows to become more prominent, so care should be taken to take image with very less shadow falling on the image. The resulting image also depends on the built in camera of the device, hence the processing and result of shadows may vary depending on the device.

## VII. EXPERIMENTAL RESULTS

The picture of the printed/written Kannada text will be taken with the help of a built in camera. And this image is then processed by converting it into a gray scale image which the application can read. It is as shown below fig1, the exact English translation of the Kannada word is displayed along with the voice output shown in fig 2.

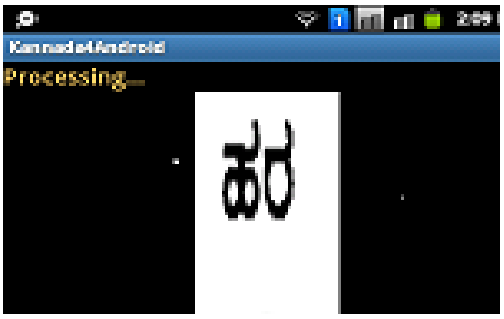


**Fig1: Processing the Image**

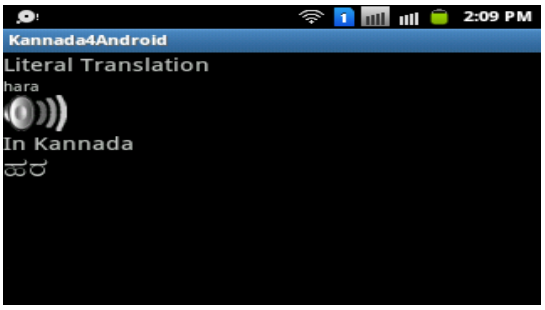


**Fig 2: English Conversion of Kannada Word**

## VIII. CONCLUSION

The paper is based on an Android application based on optical character recognition concept for normal Android device users. The implementation uses the Kohonen neural network for better accuracy and speed. The user has to make use of built in camera on the device to take a picture containing the Kannada characters

or words. The image is processed further to remove the noise, improving the contrast between the characters and the background and scaling the image. The characters are distinguished and separated from each other and individually identified through the network. With every correct identification, the network makes sure that the probability of recognizing the character increases. The characters recognized are displayed in the result along with the voice output. When the application is not able to detect the characters due to large distortion a suitable error message is displayed.

**REFERENCES**

[1]. Mark L. Murphy, "The busy coder's guide to Android development", CommonsWare, 2008.

[2]. G. Zhu and D. Doermann. "Logo Matching for Document Image Retrieval", 10th international conference on document analysis and recognition

[3]. Tariq.J, "α-Soft: An English language OCR", Computer engineering and applications (ICCEA),2010.

[4]. Yalniz, I.Z.; Manmatha. R,"A Fast Alignment Scheme for Automatic OCR Evaluation of Books", International Conference Document Analysis and Recognition (ICDAR), 2011.

[5]. R.Shukla, "Object oriented framework modeling of a Kohonen network based character recognition system",Computer communication and informatics international conference(ICCCI), 2012.

[6]. Liu, Y., R.H. Weisberg, and C.N.K. Mooers,"Performance evaluation of the Self-Organizing Map for feature extraction". Journal of Geophysical Research, 2006.