International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-5, March 2020

Machine Learning Based Braille Transliteration of Odia Language

Vinod Jha, K. Parvathi

Abstract*: Braille transliteration of natural languages is required for providing a better opportunity of learning and creating opportunities of ceceity people. It allows a bigger diaspora of non-blind teachers to have written communication with blind people. The present paper proposes a method of Braille transliteration of Handwritten and printed Odia characters automatically into Braille. The current work proposes a method of Braille transliteration of Handwritten Odia text with industry applicable accuracy. The method first preprocesses the text and then segments it into characters and then uses an SVM classifier trained on HOG features of Odia handwritten characters to predict characters and maps the predicted printable character to its corresponding Braille with a very good accuracy. The method can further be used with text to speech engines to help the blind students use this technique with refreshable Braille having audio facility to listen the same.

***Keywords*: Braille, Optical Character recognition, HOG, SVM, Unicode**

I. INTRODUCTION

India has a lion share of blind people of the world and it also has one of the highest percentages of unemployable and illiterate blind people. Therefore Bharati Braille is a language of utmost importance for the upliftment of blind people in India. But unfortunately, there is a scarcity of resources like books, schools and even teachers to teach blind people. It is very time cumbersome and complex task to write books in Braille and even costly. However with the recent advancement in the field of technology it is possible to reduce the complexity and cost of teaching blind people. The automatic conversion of Braille documents into natural languages and speech helps any arbitrary tutor to teach blind students without even knowing Braille fluently. This reduces the need of special tutors to teach Braille. The automatic conversion of documents in natural languages to Braille helps in printing books written in any natural language into Braille. It is also possible to teach the blind students using a refreshable Braille slate which can refresh the dot pattern after taking input directly from the optical character recogniser and read the text like any non-blind person would do. This conversion of Braille into natural language and vice versa is called as Braille transliteration. There are many works done in the field of Braille transliteration of languages like English, Arabic, etc. However there is scarcity of work in the field of Braille transliteration of Indian local languages like Odia. The work presented in [1] describes a realistic method of Braille transliteration of printable text in Odia. The transliterator reads the letters from the text file and converts each letter into Braille using Unicode mapping to Braille code. The Braille codes are mapped to corresponding Braille cells and then the Braille cells are organized into a standard Braille sheet. However there is still no method which can be used to make the conversion of printed sheets into Odia Braille or handwritten sheets into Odia Braille. This task has become physically realizable with considerably good accuracy because of the design of high accuracy Optical character recognizer (OCR) for Odia language. The present paper proposes a method to automatically convert handwritten characters into Braille, which can then further be used to convert handwritten texts and printed texts into Braille. The method proposes a way to convert Odia handwritten texts into Braille as follows: take an image of the text as input and convert it into a printable text format by using Odia OCR, convert characters in the printable text into the corresponding Unicode and the map the Unicode to corresponding Braille code and finally to Braille cell. The ODIA OCR is designed using NIT Rourkela dataset of handwritten ODIA characters [26] which contains 320 unique samples each of 47 different characters. The SVM with linear kernel is trained using Histogram of oriented gradient features with 8x8 block size. The highest accuracy on the OCR so obtained is 96 present. The accuracy of printable character transliteration into Braille is 100 per cent.

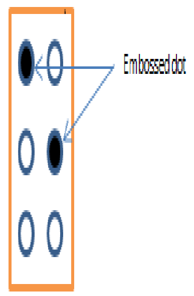


Fig.1. A Braille cell

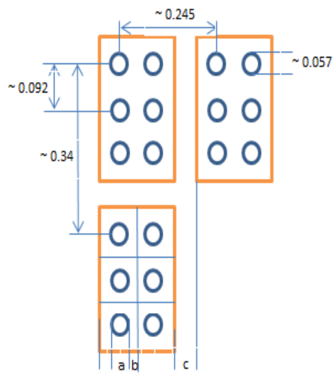


Fig.2. Braille Specification

A braille cell for Indian scripts comprises of 6 dots. By using one or more dots at a time 64 letters (symbols) in any language can be represented.

Revised Manuscript Received on March 30, 2020.
* Correspondence Author
Vinod Jha*, Assistant Professor, Department of Electronics and Telecommunication, KIIT, India.
Dr. K. Parvathi, Administrator, Department of Electronics and Telecommunication, Andhra University, Visakhapatnam, India.

Braille dots are embossed on a particular type of sheet and it is read by sensing the embossment over the braille sheet by using fingers. For the mapping purpose the braille cells are given representation. In this paper a raised dot is considered as '1' and a non-raised dot is considered as '0'. So the Braille cell shown in figure 1 is represented as '100100'(in a rowwise reading). In this way one can have all the possible braille cells represented using 6 bit binary system. All the measures provided in Fig.2 are in inches. Braille dots are embossed on a particular type of sheet and it is read by sensing the embossment over the braille sheet by using fingers. Although the thickness of the sheet is also standardized, it varies place to place. Thickness of the braille sheet is directly proportional to its life. Less the thickness, less the number of times it could be read. A standard braille sheet consists of 40 lines with maximum 25 characters each. The present work is carried out on MATLAB with the screen resolution of 96 dpi for printing standard Braille cells.

A. Related Works

There is a scarcity of Braille Transliteration in Indian regional languages. Recently in the last decade researchers have done considerable amount of work for the Braille Transliteration of Hindi, Bengali, Tamil, and Telugu etc. However, Transliteration of Odia braille is a work in progress. Braille Transliteration of Odia consists of two problems, converting Braille document to Odia and vice-versa. An Optical Braille Recognition (OBR) is achieved by first segmenting the Braille cells from a document and then converting them into Odia language characters. The present work focuses on converting scanned Odia texts into printable Braille form. In 2018[1] describes a method of Odia Braille Transliteration of printable texts. This method also describes a common algorithm for forward and reverse text transmission. It uses a Unicode of the letters and maps it to the corresponding braille cell. In another work [2] a method has been proposed for fast transliteration of Odia Braille. This method takes a printable Odia text and maps it to the corresponding Braille. Further, it searches for the number of dots present in the cell to reduce the time taken in the mapping process. Researches have been done for transcription of English [3], Tamil [3][4], Telugu, Bengali [5], Kannada etc. But they are limited to the transcription of printable texts or printed texts into Braille. Gayathri Devi et.al [6] proposed a methodology in 2018 which transliterate Tamil printed text into Braille using Deep learning with claimed accuracy of 95.7%. Another work [7] proposed a method of transliteration of Kannada Braille into printable text and then to speech on Verilog and dumped it on Xilinx Spartan 3e series FPGA. Shiv Kumar et.al [8] converted printable Tamil documents into a new file format called BRF(Braille) on JAVA eclipse SWT platform. Jie LI et.al [9] proposed a mechanism for Braille character recognition using HAAR features as support vectors. Braille to text and text to Braille transliteration has been proposed in Gujarati [10][11][12][13] and Devanagiri[1][14]. A tool has been proposed in [14] which can edit and create Devanagiri Unicode text using digital Braille typewriter. Prachi Rajarapollu et.al [15] presented the implementation of English Braille typewriter output transliteration into English texts using Spartan 3 kit on FPGA. The Odia braille transliteration problem is still wide open. For transliteration of scanned or handwritten texts, there is a need of an accurate optical character recogniser which can convert printed sheets or handwritten sheets into printable texts with acceptable accuracy so that the printable text can further be transliterated to Braille. Recently with the evolution of Deep learning the OCR accuracy has been considerably improved and researchers have achieved considerably much better accuracies. Madhuri Yadav et.al in [16] [22] proposed an OCR based on shape description of the alphabet such as histogram of oriented gradients and geometric moments. These features are used to train the classifier such as SVM and multi-layer perceptron. The work claims to achieve highest recognition rate of 96.8% on one of the dataset. Another method [17] uses 8-directional and 16-directional gradient feature (DGF) input values to train two layer fully connected back propagation feed forward network. Sobel operator is used to get the gradient features. The method claims to be approximately 96% accurate with 16-DGF which also gives much more complexity. In another method [18] Hidden Markov model is used to recognize the middle layer after segmenting the word into three layers. A new feature named pyramid histogram of oriented gradient (PHOG) is used to recognise the middle zone. The work has also compared the efficiency of using various features such as Marte-Bunke feature, PHOG, Gaber, G-PHOG and claims to get highest accuracy of 94.51% for GPHOG features. Another approach [19] of character recognition used k-mean clustering for features extraction and features are used on SVM classifier with linear kernel. The highest accuracy which was claimed with SVM classifier is 95.86% and that with Euclidean distance is 81.7%. Another approach [20] introduces a Devanagiri handwritten character dataset (DHCD) with 92000 images of 46 different classes of characters of Devanagiri script segmented from handwritten documents. It proposed a deep convolution neural network with the use of dropout and dataset increment approach and claimed an accuracy of 98.47% on the said dataset. İn [21], the proposed has used curvelet transform which approximates the curved singularities of images very well. Then KNN is trained with curvelet features which are obtained by evaluating statistics of thick and thin images by applying curvelet transform. The proposed recognition accuracy is 90% but the dataset valuate was very small. The paper in [22] introduces a new handwritten character database and also lists and compares the existing database's advantages and pitfalls. The standard databases are scanned using 300 dpi flatband scanner. The biggest database with 92000 characters has been used but the drawback is the lack of vowels in the dataset. Other datasets have relatively small number of samples per image. The Odia OCR development starting works include work by Pitabasha Pati et.al [23 [24] who proposed variable window technique based on character statistics for character segmentation from word and then used KNN to classify and recognised the character images. Another method by Bhabani Dash et.al [25] used DWT features of the printed Odia characters for character matching techniques claiming satisfactory performance in character recognition.

Tusar Kanti Mishra et.al [26] developed a standard database of Odia handwritten characters and developed an OCR for Odia numeral using hidden Markov model. The database comprises of 320 characters each of 47 Odia literals. Another method [27] made a neural network based OCR for printed Odia character. The Paper in [28] proposes a similar work for Hindi where a method is proposed to transliterate Handwritten Hindi texts into Braille. The present paper proposes a relatively less complex classifier and yet better accuracy to form an OCR for Odia handwritten characters with highest accuracy of 95.56%. The classifier is trained and tested on standard Odia handwritten dataset [26]. This method takes a scanned character of printed or handwritten documents and converts it into a printable character first and then generates a Braille equivalent of the character which can be used on a refreshable braille display or can be printed on special braille sheets using Braille embosser. All the works has been carried out on MATLAB platform with Nirmala UI Semilight font for display of all characters.

II. METHODOLOGY

The paper deals with the Braille conversion of Odia handwritten characters. This proposed method can be used for Braille transliteration of any Odiya texts as given in the following flow chart in fig.3.

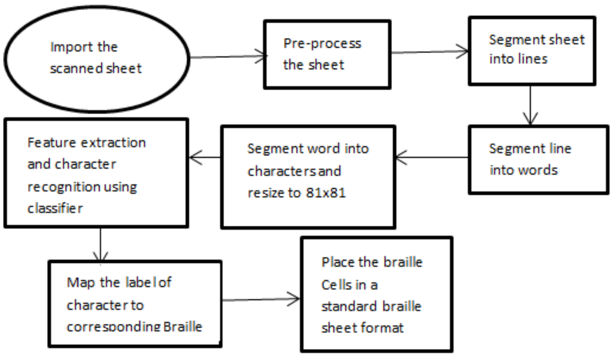


Fig.3. Flow chart of the algorithm

The resolution of scanning should not be less than 300 dpi to have good accuracy. Each character is segmented by using connected components. The segmented characters are then recognised using the SVM classfier. The recognised class of the character is displayed and it is mapped to the corresponding unicode(hex code) of the letter. This step can be used to create an OCR as well for further conversion of text to speech. Further the unicode of the recognised character is mapped to correspondning Braille code which itself is a binary code. Corresponding to this binary code, braille cells are generated.
The table used for Braille mapping is shown Table 1.

III. THE HOG FEATURE VECTOR AND THE CLASSIFIER

A standard database [26] of 15040 Odia character images is used to train and test an SVM classifier using 'one vs one' multiclass classification scheme. The database comrpised of 320 samples each for 47 characters. The matras are excluded from the database. The database is divided into training and testing images at the ratio of 4:1. The 'one vs one' SVM classifier is trained on 47 characters using 47*46/2 number of binary SVM classifiers using linear kernal. The classifier's hyper-parameter are tuned to get better training accuracy. The 10-fold hyper-paramter optimazation is achived by dividing training set into 10 parts with each part being utilised for training and testing. Each portion of the original training set is further divided into training and testing sets at 9:1 ratio. The optimized SVM classifier is showing testing accuracy of more than 96 percent.

Table I. Odia character to Braille code mapping

Letter	Hex Code	Braille dot pattern	Letter	Hex Code	Braille Dot pattern
ଅ	BO5	100000	ଧ	B27	11011
ଆ, ା	B06, B3E	10110	ନ	B28	110110
ଇ, ି	B07, B3F	11000	ପ	B2A	111010
ଈ, ୀ	B08, B40	110	ଫ	B2B	1110
ଉ, ୁ	B09, B41	100011	ବ	B2C	101000
ଊ, ୂ	B0A, 42	101101	ଭ	B2D	10100
ଏ, େ	B0F,B47	1001	ମ	B2E	110010
ଐ, ୈ	B10, B48	10010	ଯ	B2F	110111
ଔ, ୌ	B14, B4C	11001	ଳ	B32	101010
କ	B15	100010	ଵ	B35	101011
ଖ	B16	10001	ଶ	B36	110001
ଗ	B17	111100	ଷ	B37	111011
ଘ	B18	101001	ସ	B38	11010
ଙ	B19	10011	ହ	B39	101100
ଚ	B1A	110000	ଁ	B01	10
ଛ	B1B	100001	ଂ	B02	101
ଜ	B1C	11100	ଃ	B03	1
ଝ	B1D	111	୍	94D	10000
ଞ	B1E	1100	ଽ	B3D	1000
ଟ	B1F	11111	ଡ଼	B5C	111101
ଠ	B20	11101	,	2C	1000
ଡ	B21	111001	;	3B	1010
ଢ	B22	111111	:	3A	1100
ଣ	B23	10111	SPACE	20	0
ତ	B24	11110	?	3F	1011
ଥ	B25	110101	!	21	1110
ଦ	B26	110100	|	7C	1101

The classifier is trained on HOG features of the training images. The Histogram of oriented gradients works well for detection of objects beacuse it represents the image in forms of gradients with less number of features. The gradients remove lots of unnecesary information like constant coloured background. The gradients stores the information of the shape very well and when these features are used with learning algorithm like SVM, the observed accuracy is remarkable.

To find the HOG features of an image, the gradients are found for all the pixels along rows and columns. Using these values the magnitude and phase of gradients are found. Now the image is divided into 8x8 cells. A histogram of 9 bins between 0 to 180 degrees is made using phase of the gradients. Suppose for a particular pixel in a cell the magnitude of gradient is 40 and angle is 60, so 40 is put in the 4th bin which corresponds to 60 degree angle.
Now suppose another pixel in the same cell has a gradirnt magnitude of 30 at an angle 50 degrees then this magnitude will be divided into 3rd and 4th bin in equal proportion. The new content of 3rd bin will be 15 and 4th bin will be 40+15=65. The below figure, shows how the feature vector is created after finding out magnitude and angle of gradients in a cell[28].

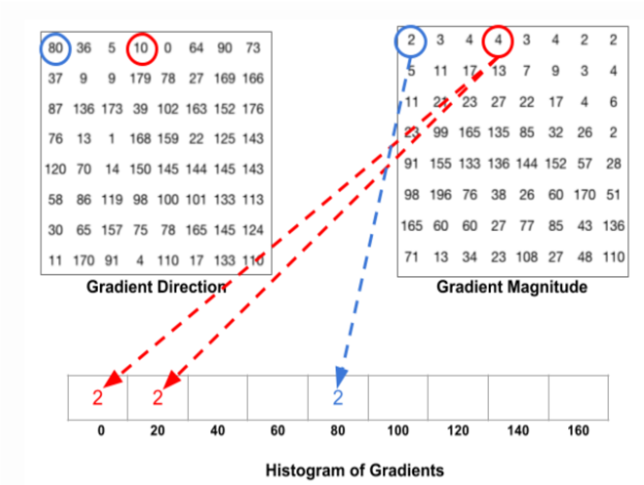


Fig.4. Preparing HOG feature vector for a 8x8 image cell

IV. RESULT ANALYSIS

The following table shows the result of the SVM classifier in 20% of the database images used for testing. The result seems overwhelming with most of the letters getting accurately recognised.

Table II. Classifier prediction on 20% test letters of the database

Sl. No.	Odia letter	%accuracy	Sl. No.	Odia letter	%accuracy
1	ଅ	100.0000	25	ଡ	100.0000
2	ଆ	100.0000	26	ଢ	100.0000
3	ଇ	100.0000	27	ଣ	100.0000
4	ଈ	100.0000	28	ତ	100.0000
5	ଉ	100.0000	29	ଥ	100.0000
6	ଊ	100.0000	30	ଦ	100.0000
7	ଋ	100.0000	31	ଧ	100.0000
8	ୠ	100.0000	32	ନ	100.0000
9	ଏ	100.0000	33	ପ	100.0000
10	ଐ	100.0000	34	ଫ	100.0000
11	ଓ	100.0000	35	ବ	100.0000
12	ଔ	100.0000	36	ଭ	95.3125
13	କ	100.0000	37	ମ	100.0000
14	ଖ	100.0000	38	ଯ	95.3125
15	ଗ	100.0000	39	ର	95.3125
16	ଘ	100.0000	40	ଲ	100.0000
17	ଙ	100.0000	41	ଳ	100.0000
18	ଚ	100.0000	42	ଶ	100.0000
19	ଛ	100.0000	43	ଷ	100.0000
20	ଜ	100.0000	44	ସ	100.0000
21	ଝ	100.0000	45	ହ	100.0000
22	ଞ	92.1875	46	ଡ଼	95.3125
23	ଟ	100.0000	47	କ୍ଷ	95.3125
24	ଠ	100.0000			

The various stages of a handwritten letter being transliterated to Braille are shown in the figures below:

Fig 5.a.

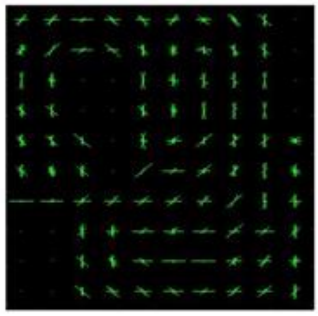


Fig 5.b.

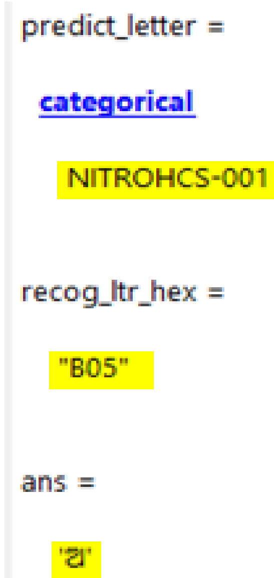


Fig 5.c.

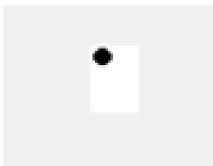


Fig 5.d.

Fig5. a). Odia Letter 'A', b).Its HOG features, c). Prediction using the classifier and mapping to corresponding hex code and d.)Braille transliterated letter

Above result is obtained by taking an arbitrary test image and performing various stages of the methodology. Similarly the same procedure is repeated on a number of independent images and results were satisfactory. Another result is show in the figures below:

Fig 6.a.

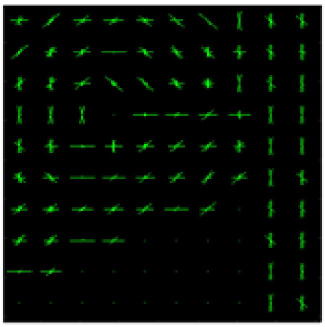
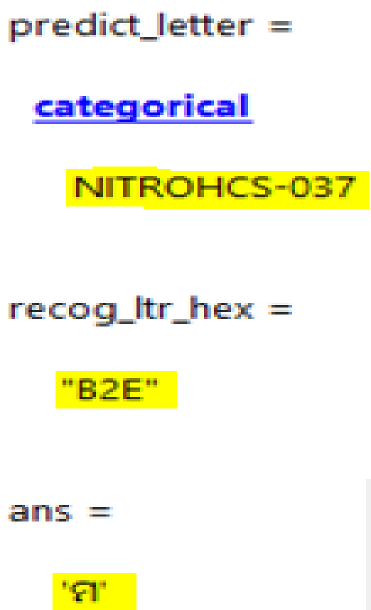


Fig 6.b.



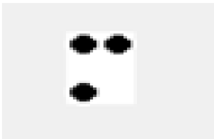


Fig 6.c. Fig 6.d.

Fig6. a). Odia Letter 'MA', b).Its HOG features, c).

Prediction using the classifier and mapping to corresponding hex code and d.)Braille transliterated letter

The method can be applied to Odia words, sentences and paragraphs by doing lines and words segmentation followed by character matching as above. One example of application of the above method on a handwritten word is show below:

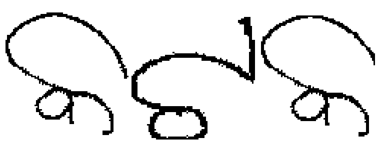


Fig. 7 a sample handwritten Odia word

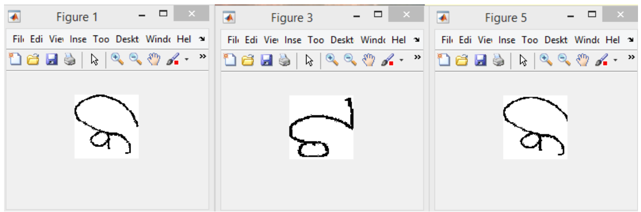


Fig. 8. Segmentation of the word into characters

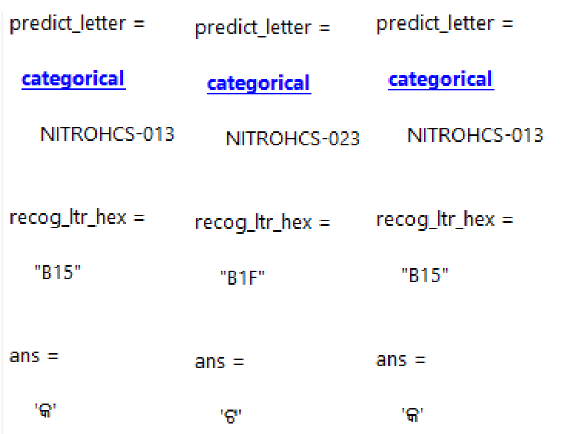


Fig. 9. a, b &c. Recognizing segmented characters

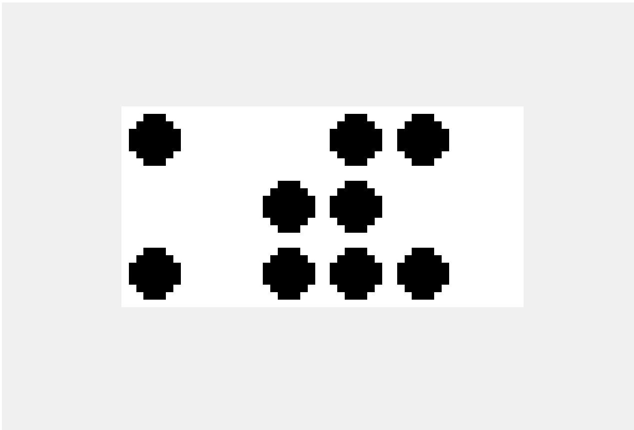


Fig. 10. Braille transliteration of the above handwritten word

V. CONCLUSION AND FUTURE WORK

The work implemented The results are shown for characters which are not used in training the classifier and the results suggest that the method is practically implementable. The accuracy of the classifier on the testing data is approximately 99 per cent. It also worked very well on real time texts (which are not part of database). As suggested earlier, this method can be integrated with line segmentation and word segmentation to get the Braille transliteration of arbitrary texts in Odia. Further the classifier may be improved and made to consider Odia numerals as well. Storing the Braille dots as 0's and 1's has its own advantage that it takes only 6 bits to store a letter. It will take much more space if numbers are used to represent the same as been done in many works. Further the algorithm is improved upon the speed of mapping the predicted class to the corresponding Braille.

This work is intended to be used with refreshable Braille to assist non-blind tutors who do not know Braille, teach blind students. This method can be further associated with text to speech engines to provide audio assistance along with Braille texts to help the blind students further.

REFERENCES

1. Vınod Jha, K.Parvathi "Complete Forward and Reverse Text Transcription of Odia Braille and Hindi Braille", : Progress in Advanced Computing and Intelligent Engineering, Springer, sequence no.12,2018
2. Bijet Maynoher Samal, K.Parvathi, Jitendra Kumar Das, "A bidirectional text transcription of Braille for Odia, Hindi, Telugu and English via image processing on FPGA". International Journal of Research in Engineering and Technology, Volume: 04 Issue: 07, July-2015
3. S.Padmavathi,Manojna K.S.S, Sphoorthy Reddy .S and Meenakshy.D, "conversion of Braille text in English Hindi and Tamil languages". IJCSEA, Vol.3, No.3, June 2013
4. Shiva Kumar H R, A.G.Ramakrishnan,"tool that converted 200 Tamil books for use by blind students", 2013
5. Santanu Halder, Abul Hasnat, Amina Khatun, Debotosh Bhattacharjee, Mita Nasipuri , "Development of a Bangla Character Recognition (BCR) System for Generation of Bengali Text from Braille Notation". IJITEE, ISSN: 2278-3075, Volume-3, Issue-1, June 2013
6. Dr. G.Gayathri Devi "A Braille Transliteration on Tamil Vowels and Consonants Text Image",IJAER, Volume 13, Number 11 (2018) pp. 8907-8912
7. S.R.Rupanagudi, S.Huddar, V.G. Bhat, Suman S. Patil, Bhaskar M.K,"Novel Methodology for Kannada Braille to Speech Translation using Image Processing on FPGA". Advances in Electrical Engineering(ICAEE), International Conference 2014.
8. Shiva Kumar H R, A.G.Ramakrishnan,"tool that converted 200 Tamil books for use by blind students".
9.] Jie Li et al.(2010). "Optical Braille Recognition with Haar Wavelet Features and Support-VectorMachine." International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE).
10. Nikisha B. Jariwala, Bankim Patel "Conversion of Gujarati Text into Braille: A Review", IJIAC, Volume 4, Issue 1 January 2015
11. Nikisha Jariwala, Bankim Patel, "A System for the Conversion of Digital Gujarati Text to Speech for Visually Impaired People", Speech and Language Processing for Human-Machine Communications, pp 67-75
12. Hardik A .Vyas et.al. "Transliteration of Braille character to Gujarati text-Model", IJERT, ISSN: 2278-0181, vol.7, issue 7, July-2018
13. H.A.Vyas et.al. "Transliteration of Braille Character to Gujarati Text The Application", IJCSE, 7(2), 701-705,Feb-2019
14. Diwakar Mishra, Bhartendu Pandey, " Editing Unicode Devanagari Text through Digital Braille Typewriter". 22nd International Congress of Vedanta, Dec .2015.
15. Prachi Rajarapollu et.al, "FPGA based Braille to text & speech for blind persons", IJSER, vol.4,Issue 4, April-2013
16. Madhuri Yadav et.al. "Design of CNN architecture for Hindi Characters", ADCAIJ, Regular Issue, Vol. 7 N. 3 (2018), 47-61
17. Dayashankar Singh et.al. "Analysis of handwritten hindi character recognition using advanced feature extraction technique and back propagation neural network", IJCA, vol. 97
18. P.P. Roy et.al.. "HMM based indic handwritten word recognition using zone segmentation", Pattern Recognition,60(2016), pp 1057-1075
19. Akanksha Gaur et.al., "Handwritten hindi character recognition using K-means clustering and SVM",ETTLIS,IEEE, 2015
20. Shaliesh Acharya et.al. "Deep learning based large scale handwritten Devanagari character recognition", SKIMA-2015, IEEE
21. Gyanendra K. Verma et.al. "Handwritten hindi character recognition using curvelet transform", ICISIL, 2011, pp. 224-227
22. Madhuri Yadav et.al."A new database of Hindi handwritten characters",IJIACS, ISSn 2347-8616, vol. 7, Issue 4, April-2016
23. Peeta Basa Pati et.al, "Machine recognition of printed Odiya characters",ICIT, 2000
24. Peeta Basa Pati et.al, "Machine recognition of printed Odiya texts", A report, 2004
25. Shibashis Pradhan et.al, "Odia offline character recognition using DWT features", IOSR-JECE, PP 31-37, 2016
26. Tusar Kanti Mishra, Banshidhar Majhi, Pankaj K Sa and Sandeep Panda, " Model based odia numeral recognition using fuzzy aggregated features", Front. Comput. Sci., Springer, 2014, 8(6): 916–922, DOI 10.1007/s11704-014-3354-9.
27. Soumya Mishra et.al, "Oriya character recogntion using neural network", IJCCT, vol.2 , issue 2,3&4, 2010
28. Vinod Jha, K. Parvathi, "Braille Transliteration Of Hindi Handwritten Texts Using Machine Learning For Character Recognition",International Journal of Scientific and Technology Research, Vol.8, Issue 10, October 2019
29. https://lilianweng.github.io/lil-log/2017/10/29/object-recognition-for-dummies-part-1.html

AUTHORS PROFILE

Vinod Jha He is a PhD scholar and assistant professor in KIIT, School of electronics school. He did his B.Tech from KIIT in Electronics and Telecommunication and M.Tech in signal processing from IIT Guwahati. He has 9 years of experience in teaching and research. He also holds degree of Post graduate diploma in Data science from IIIT Bangalore and Upgrad. Being a faculty, he has taught various subjects like Signal & System, Digital Signal Processing, Analog Electronics, Digital Electronics, Microprocessor, Analog Communication Technique, Digital Image Processing etc. He has published 2 papers in his PhD work so far which is application of image processing and Machine learning in Braille transliteration. Orcid ID: https://orcid.org/0000-0001-7774-4508
Scopus ID: 57200731507

Dr. K. Parvathi Being an administrator, academician, she has 25 years of teaching, research, and 7 years of administrative experience. She did her Bachelors from Osmania University, Hyderabad and Masters, Ph.D.(Engineering) from Andhra University, Visakhapatnam. For the past 25 years, she is dealing with courses like Analog & Digital communication Techniques, Digital Signal Processing, Digital Image Processing, Optical Communication & Satellite Communication. Her research interests are Signal & Image Processing, Computational Intelligence. She has published research papers in 25 International Journals, 30 International Conferences. She also produced 2 PhD scholars and 4 others are continuing under her guidance. She is a life member of various professional bodies like IET, ISTE.
Scopus Author ID: 36662827800
Orcid ID: http://orcid.org/0000-0002-8988-9576