

Transliteration of Braille Character to Gujarati Text – The Application

H.A. Vyas^{1*}, P.V. Virparia²

¹Babu Madhav Institute of Information Technology, Uka Tarsadia University, Tarsadi, Surat, Gujarat, India

²G. H. Patel Post Graduate Department of Computer Science, Sardar Patel University, Vallabh Vidyanagar, Anand, Gujarat, India

*Corresponding Author: hardik.vyas@utu.ac.in, Tel.: +91-98256-58143

DOI: <https://doi.org/10.26438/ijcse/v7i2.701705> | Available online at: www.ijcseonline.org

Accepted: 17/Feb/2019, Published: 28/Feb/2019

Abstract — Technology has a significant contribution in the digital communication for visually impaired persons. There is a significant need transliteration of Braille Characters into regional language. The “Software for Transliteration of Braille Character to Gujarati Text using Python” has three key factors keep in mind. First there is digital communication using technology to provide alternative means of reading and writing. Next, technology can help in the production of digital content converted from Braille to Gujarati or vice versa. A third factor for technology is to provide improved access to information like import, export and saving the digital content. It is sometimes quite difficult to be certain whether a technology or system will meet particular needs or not. This paper is about developing a prototype using Python Programming Language, because Python is a suitable language for both learning and real world programming. Python is an extremely potent and high level object-oriented programming language. We have provided various techniques describe as a literature to develop the tool and also identified the challenges that might be faced at the time of recognition and conversion

Keywords — Braille translation, Braille display terminal, Braille character conversion modules, Unicode recognition using Python, Braille cells, Braille to Gujarati

I. INTRODUCTION

Information in written form plays an undeniably important role in our daily lives. The recording and then retrieving information is vital whether we want that for education, entertainment or any type of information exchange. In that, the visually impaired individuals are in loss.

Addressing this need, the most widely adopted writing convention among visually impaired people is Braille. With about 3 and a half million in numbers, India hosts most blind people on the planet visually challenged people to experience immense difficulties in accessing data from documents [1].

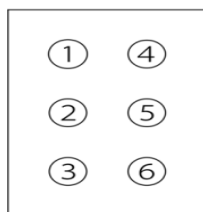


Figure 1

The way of writing Braille by using the patterns in the form of raised dots to inscribe the characters on a sheet of paper. It therefore, allows visually – impaired people to read and write using touch instead of vision. The blind

people can take a part in literate culture using Braille has become the pre-eminent tactile alphabet. As described in Figure-1, there are six dots which are two wide and three tall in Braille. There are 64 possible characters as any of the dots can be raised [2].

Every single of 64 cells can be read according to the language it is used to depict; hence Braille cells have global application for any language. Since the characters of every language are different- be it alphabets, punctuations, or numbers, the Braille codes are made according to them.

Even a group of letters including whole words can be depicted by using Braille characters.

Then this text can be transliterated into Braille for output to Braille printer or directed to special Braille output devices [3].

II. LITERATURE SURVEY

The basic algorithms for computerized Braille translation have been around since the late 1960s [4]. In the 1980s, personal electronic Braille note-takers entered the marketplace and, as a result, Braille-to-print translation software began to be developed along with the older print-to-Braille technology. Earlier, this representation was only applicable in the English language, but now it is being used for different regional languages. During the evolution

of computer programming the mechanism to type the Braille characters through the keyboard, Braille documents are pre-processed to enhance the dots and reduce the noise. The typed Braille character is mapped to the character set [5]. In the conversion of scanned Braille documents to text in English, Hindi and Tamil languages is emphasized. The converter takes idiosyncrasies into consideration and attempts to make the translation between *Devanagari* and *Bharati Braille* as accurate as possible [6].

III. PROPOSED TECHNOLOGY

Braille is a code, not a language and these Braille codes are used for representing Braille character in any language like English, Hindi, Gujarati, Odia, Bengali, Kannada, Urdu. For converting the Braille document to Gujarati text, the input is taken in two different formats. In the first method, the Braille character is accepted as a series of numbers typed through the keyboard and in the second method a softcopy (.txt or .doc format) Braille document is taken as input. The Braille character after extraction in each case is then mapped with the corresponding Gujarati character.

3.1 Gujarati Script



Figure 3

Gujarati script is derived from Devanagari script and is descended from Sanskrit. The Gujarati alphabet has an overall 75 distinct legitimate and recognized shapes, which mainly includes 59 Characters and 16 diacritics. A total of fifty-nine characters are split into 36 consonants, having 34 singular and 2 compound ones although not in the lexical sense, having ornamented sounds with 10 pure sounds as vowels and 10 digits of numbers. Thirteen vowel and 3 other characters made sixteen diacritics. It is to be noted that the Gujarati script moves from left to right. The vowels are called Swar and consonants are called Vyanjan. Specially modified symbols called Maatras which correspond to each vowel make the Gujarati Language; they in turn, are added to consonants for changed sounds. The Modifiers are marked on top or at right bottom or at a consonant's bottom. The Braille cell for Gujarati text can be generated with the help of the mapping table (Figure 2) where a binary equivalent dot pattern for the corresponding Gujarati text is shown. The raised dots are read as '1' and the non-raised dots are read as '0' [7].

3.2 Python

Python is a suitable language for both learning and real-world programming. Guido Van Rossum created Python and it's a high level object-oriented programming language. Python supports a dynamic type system and automatic memory management and has a large and comprehensive standard library [8]. Python interpreters are available for many operating systems. Python's status as the fastest-growing programming language is being fuelled by a sharp uptick in its use for data science. According to the TIOBE index, Python is the 4th most popular programming language out of 100; with the rise of Ruby on Rails and more recently Node.js, Python's usage for the web development more so due to fragmented MVC ecosystem it has [9].

3.3 Input Format

અ	આ	ઇ	ઈ	ઉ	ઊ	એ	ૈ	ઓ	ઔ
□	□	□	□	□	□	□	□	□	□
ક	ખ	ગ	ઘ	ઙ	ચ	છ	જ	ઝ	ઞ
□	□	□	□	□	□	□	□	□	□
ટ	ઠ	ડ	ઢ	ણ	ત	થ	દ	ધ	ન
□	□	□	□	□	□	□	□	□	□
પ	ફ	બ	ભ	મ	ય	ર	લ	વ	ળ
□	□	□	□	□	□	□	□	□	□
શ	ષ	સ	હ	ણ	ક્ષ	ઞ			
□	□	□	□	□	□	□	□	□	□
અં	અઃ	અઃ	~	અં					
□	□	□	□	□	□	□			

Figure 2

The objective of software prototype is to accept the input as a text written in Braille script and convert it into Gujarati script. For many blind people, Braille is their natural medium for literacy. Regular QWERTY keyboard might not be useful for them to enter information to the computer; it can be achieved by Braille keyboard or specific keys on a QWERTY keyboard. In this paper combination of SDF and JKL (six keys) are used to enter data in to Braille code as shown in Figure 3.

3.4 Execution

This paper is about the editor which is developed using Python programming language, which accepts Braille Text as input. Python first compiles source code (.py file) into a format known as byte code. Compilation means a translation step with lower-level byte code and a representation of source code that is platform independent. The Compiled codes are saved as .pyc files and can be regenerated on updating source or whenever required.

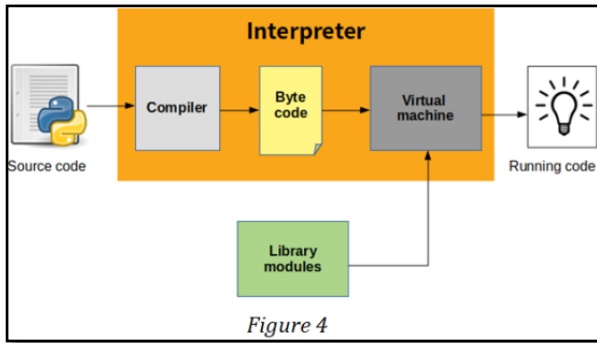


Figure 4

As shown in Figure 4, the bytecode (.pyc file) is loaded into the Python runtime and interpreted by a Python Virtual Machine (PVM), which is a piece of code that reads each instruction in the bytecode and executes character recognition operation is indicated. Byte code compilation is automatic, and the PVM is just part of the Python system which truly runs scripts. Technically, it's just the last step of what is called the Python interpreter.

The interpreter has to obligatorily convert source code into the machine code and have to pull the runtime libraries whenever an interpreted program is made to run. But the process of conversion makes a program to run slower than the one written in the compiled language. Python language enhances its performance by compiling to bytecode (.pyc) files on the very first execution of a file. This improves substantially the execution of the code next time the module is imported or executed.

3.5 Character Recognition

The proposed research model accepts a combination of Braille numbered dots as shown in Figure 5. Python code will recognizes entered Braille character with its equivalent Gujarati character as earlier shown in Figure 2. There are total 6 dots in single cell. So, total of 64 characters can be written. But Gujarati language has a total 75 characters. So in Gujarati Braille, there are some characters (consonant and vowels) that are identical to the numerals when represented in Braille. So some assumptions are considered while writing Gujarati Braille. This character is the smallest possible component of a text. Python compiler has to identify its UTF8 equivalence character to display Gujarati script. A code point represents an integer denoted in the base 16 often. Hence a Unicode string is basically a series of code points numbered from 0 to 0x10ffff. Python supports a translation package based on web service provided by Google, Youdao, Baidu, iciba for recognition [10]. For example;

```
$ pip install py-translate
$ translate enzh-gu<<< 'kemcho'
output: કેમચો
```

3.6 Representation

```
def save():
    global path
    global saveneed
    name=tkFileDialog.asksaveasfilename(defaultextension=".txt",
        filetypes=[("Text files",".txt"),
            ("Word files",".doc")],
        initialdir="dir",
        title="Save as")

    path=name
    saveneed = False
    with open(path,"w") as data:
        dt = ent.get("1.0",END)
        index = dt.find('@')
        dt = dt[:index]+dt[index+1:]
        dt = dt.strip()
        data.write(dt.encode('utf-8'))
```

Snippet 1

The characters from a file executed using Unicode array of Gujarati script by extracting each cell from the Braille text file. Following code snippet - 1 will convert extracted Braille cell to its Unicode. Then store all Unicode of the text into an array and Retrieve element (Braille cell) from the array. Identify the code of element if the code is of Gujarati letter (Consonant and Vowel) then map the code with the Mapping table and according to the code recognition is done for the cell. If the code is of Digit Identifier then retrieve next code from the array until space is encountered and map the code with the mapping table and according to it recognize the digits. If the code is of Punctuation then match the code to the mapping table and recognize the Braille cell. This technique will work as a mapping technique.

3.7 Features

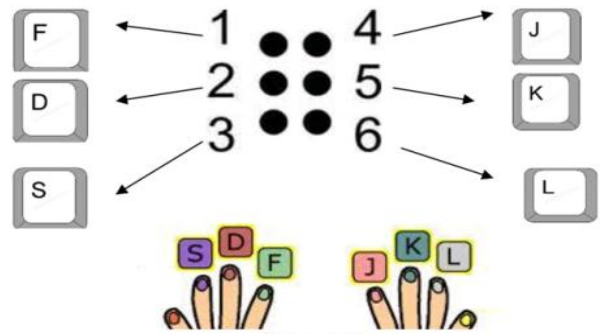


Figure 5

The proposed Software is least expensive which would provide the advantage over the commercially available systems. The User is able to save the documents and convert the same into Braille. Also, the software allows taking Braille Print through standard Braille embosser. Software has a friendly interface for visually impaired people. They should be able to easily using 6 key features (FDS JKL) Extracts the contents of “.doc”, “.txt” link pages also. Multiple character information can be transmitted with high accuracy. It provides the following menus with easily understood layout:

```

menubar = Menu(root)
filemenu = Menu(menubar)
filemenu.add_command(label="New",command=newfile);
filemenu.add_command(label="Open",command=opens);
filemenu.add_command(label="Save",command=directsave);
filemenu.add_command(label="Save as",command=save);
filemenu.add_command(label="Close",command=closeprog);
menubar.add_cascade(label="File",menu=filemenu);
editmenu = Menu(menubar)
editmenu.add_cascade(label="Copy",command=copy)
editmenu.add_cascade(label="Paste",command=paste())
editmenu.add_checkbutton(label="Gujrati Show",variable=toggle,command=change)
menubar.add_cascade(label="Edit",menu=editmenu);
root.config(menu=menubar);

```

IV. GUI OF SOFTWARE

The menu driven and easy to use layout make this software more effective. Python itself is a powerful language for performance oriented execution which makes software more reliable. Software has menu like New, Open Save, Save As and Close as submenu in File Menu and Copy, Paste and Gujarati Show as submenu in Edit Menu (Figure 6).



Figure 6

Figure 7 shows the sample input given in Braille Script and equivalent Gujarati Script.



Figure 7

V. CONCLUSION AND FUTURE WORK

The developed adaptive recognition algorithms can be extended not only for other Indian languages but also for the rest of the Braille languages in the world by modifying the front end of the developed system. This research is focused mainly on developing a system that reduces the difficulty in translating Braille into Gujarati document and introducing a cognitive system which can give the visually impaired people a sense of haptic feeling. This system cost effective and the satisfaction of the user is tremendous. However, this can be the ultimate solution for the blind people for Transliteration of Braille Character to Gujarati Text like other languages into Braille script. In future, the research may be the focus on Braille translation using speech in the Gujarati language.

REFERENCES

- [1] Chitte PP, P. Y. (2015 Jan). "Braille to text and speech for cecity persons". International Journal of Research in Engineering and Technology, Vol. 4(1):263–8.
- [2] Acharya. (n.d.). Disabilities. Retrieved from International Journal of Scientific Research in Computer Sciences and Engineering (ISSN: 2320-7639)
- [3] Hardik Vyas, P. V. Virparia (December 2014). "Optical Gujarati Braille recognition: A Review". International journal of emerging technologies and applications in engineering, technology and sciences (ij-eta-ets).

- [4] R. A. J. Gildea, G. H. (March 1973). "Computerised Braille production. International Workshop in Munster (Germany). Munster, Germany": Rechenzentrum der University.
- [5] Padmavathi S, M. K. (2013 Jun). "Conversion of braille to text in English, Hindi and Tamil languages". International Journal of Computer Science and Applications, 3(3):19-32.
- [6] Wajid M, A. M. (2011). "Imprinted braille character pattern recognition using image processing techniques". IEEE International Conference on Image Information processing. Shimla, India.
- [7] Hardik Vyas, P. V. Virparia (July 2018). "Transliteration of Braille Character to Gujarati Text - The Model". International Journal of Engineering Research & Technology (IJERT), V.07,I.07.
- [8] Kuhlman, Dave. "A Python Book: Beginning Python, Advanced Python, and Python Exercises". 23 June 2012.
- [9] TIOBE Software Index (2011). "TIOBE Programming Community Index Python"
- [10] Hardik Vyas, P. V. Virparia (New Delhi, 2015). "Gujarati Braille Text Recognition: A Design Approach. Advances in Intelligent Systems and Computing" - Springer, 31-40.

Authors Profile

Hardik A. Vyas is working as an Assistant Professor in Uka Tarsadia University. He has 11.9 years of teaching experience. He completed his MCA in 2007 from Veer Narmad South Gujarat University, Surat, Gujarat. He is pursuing Ph.D. from Sardar Patel University, Vallabh Vidyanagar, Gujarat. He got research grant from Government of Gujarat for his project. He published a book. Book title is "Introduction to Computers". He is also a convener of a book. Book title is "Sambharnu Singodnu". He published and presented 12 research paper in International Journal and conferences. He is an editorial review board member in several journals/magazines.



Paresh V Virparia joined the Department of Computer Science of Sardar Patel University, Vallabh Vidyanagar in 1989 and currently working as a Professor. He was a Director of Computer Science Department of Sardar Patel University during 2012-2017. He completed his MCA in 1989 from Sardar Patel University and Ph. D. in 2002 from Sardar Patel University. TEN research scholars have completed their Ph.D. (Computer Science) under his guidance. Currently, FOUR students are doing their Ph. D. under the guidance of him. Also, three students have completed their M.Phil. (Comp. Sc.) under his supervision. His publications include one book, two chapters in book and 40 papers in International Journal, 16 papers in National Journals and 47 papers in national conferences/seminars. His research interests include the areas of Computer Simulation & Modeling, Data Mining, Networking and IT enabled services. He is an editor and editorial review board member in several journals/magazine.

