

CONVERSION OF ENGLISH CHARACTERS INTO BRAILLE USING NEURAL NETWORK¹

Assistant Prof. Dr. Mohammed Y. Hassan²

Ahmed G. Mohammed³

Abstract

The Braille system is a method that is widely used by blind people to read and write. Braille generally consists of cells of raised dots arranged in a grid. The presence or absence of dots can be sensed by the blind people's fingertips to give them the coding for the symbol.

The electronic revolution is changing the way Braille is produced, stored and retrieved, making it easier to use in the work place. All kinds of materials can be put into Braille, from bank statements, bus timetables, maps to music. But it is still difficult to produce an error-free Braille for complex materials. In this paper, the ability of the neural networks will be tested to be used for translating scanned text pages, books or lectures from English language into Grade I Braille; so that blind people can deal with it.

An artificial neural network is designed with minimum structure and tested to convert the English characters into grade I literary Braille code. English characters will be assumed to be affected by noise of mean variant between 0 and 0.4. The output of the N.N can be stored in a data file that can be sent into a Braille printer or a Braille display.

Keywords: Neural Network, conversion, Braille, English characters.

الخلاصة

نظام برايل هو الأسلوب الذي يستخدم على نطاق واسع من قبل المكفوفين لغرض القراءة والكتابة . تتألف طريقة برايل عموماً من خلايا نقاط رفع مرتبة في شبكة . إن وجود أو عدم وجود نقاط يمكن أن يتحسسها الشخص الأعمى بواسطة رأس الأصبع لمنحه الترميز للرمز . غيرت الثورة الإلكترونية طريقة إنتاج برايل وتخزينها واستردادها ، مما يجعل من السهل استخدامها في مكان العمل . ويمكن لجميع أنواع المواد أن توضع في طريقة برايل ابتداءً من البيانات المصرفية إلى الجداول الزمنية للحافلة وخرائط الموسيقى . ولكن لا يزال من الصعب أن تنتج طريقة برايل خالية من الأخطاء للمواد المعقدة .

سيتم في هذا البحث اختبار قدرة الشبكات العصبية المستخدمة لترجمة صفحات النص التي تم مسحها ضوئياً أو كتب أو محاضرات من اللغة الإنكليزية إلى برايل من النوع الأول بحيث يمكن للمكفوفين أن يتعاملوا معه . تم تصميم شبكه عصبيه اصطناعية بالحد الأدنى للهيكل وتم اختبارها لتحويل أحرف اللغة الإنكليزية إلى برايل من النوع الأول الأدبيه . تم افتراض أن الأحرف الإنكليزية تتأثر بضجيج بمتوسط متغير بين (0 و 0,4) . ويمكن لخرج الشبكه العصبيه تخزينها في ملف بيانات والتي يمكن إرسالها إلى طابعة برايل أو عرضها على شاشه.

¹ This paper was presented in the Engineering Conference of Control, Computers and Mechatronics On Jan. 30- 31/2011, University of Technology.

² Control and Systems Eng. Dept. /University of Technology

³ Rimal Medical Services Company

1- Introduction:

The Braille system is a method that is widely used by blind people to read and write; Figure (1). Braille generally consists of cells of six raised dots arranged in a grid of two dots horizontally by three dots vertically. The dots are conventionally numbered 1, 2, and 3 from the top of the left column and 4, 5, and 6 from the top of the right column, as shown in Figure (2). The presence or absence of dots gives the coding for the symbol [1]. Today, different Braille codes (or code pages) are used to map character sets of different languages to the six bit cells. Different Braille codes are also used for different uses like mathematics and music. In addition to simple encoding, modern Braille transcription uses contractions to increase reading speed; therefore; Braille method is classified as a nonlinear writing [1].

Many authors have dealt with the use of Neural Networks for handprint OCR and character recognition using Neural Network[2][3][4]. These researches include the recognition of Arabic Hand Written Characters [5][6]. Furthermore, many authors have discussed detailed descriptions of methods for converting English letters into Braille and display characters on display terminals [7] [8]. Also, the quality testing of Braille printed paper was also investigated [9].



Figure (2): Braille cell.

2- Braille System:

Braille system was presented by Louis Braille in 1834. Common Braille characters are based on a 6-dot cell having two columns of three dots. Dot height is approximately 0.02 inches (0.5 mm). The horizontal and vertical spacing between dot centers within a cell is approximately 0.1 in (2.5 mm). The blank space between dots on adjacent cells is approximately 0.15 in (3.75 mm) horizontally and 0.2 in (5.0 mm) vertically. A standard US Braille page is 11.5 by 11 inches and typically has a maximum of 25 lines and 40 to 42 Braille cells per line [1].

Some 6 dot Braille cells have numerous meanings. Numbers, capital letters, and many symbols require more than one cell to produce 6-dot Braille data. There are four Braille codes currently used in the United States; namely: the Literary Braille Code, the Nemeth Braille Code, Computer Braille Code (CBC), and the Music Braille Code. There are differences in the Braille alphabet codes used in various English-speaking countries, which prevent the exchange of Braille materials. Other Braille codes are presently under development, some require the use of 8 dots or other raised symbols [10].



Figure (1): Picture of reading Braille.

2.1 Types of Braille Codes:**A- Literary Braille Code: [11]****1) Grade I Braille**

The basic code is called grade I Braille; it is a direct substitution of normal print letters for letters from the Braille alphabet which can be read by all Braille users, and it is the first stage of learning to read Braille.

2) Grade II Braille

Grade II is a shorter form which makes reading and writing Braille much faster. Grade II Braille consists of contractions added to the combinations to represent letter groups such as 'the' and 'for'. The contractions are too numerous to list here. Grade II should be used for longer signs i.e. an opening plaque or site description. Grade II is also used widely in books, magazines and leaflets. It occupies less space than Grade I Braille but all Braille is bulky. For example, one A4 printed page would need approximately two and a half pages to reproduce in Braille.

B- Nemeth Braille Code

It is used for mathematics and science material which contains symbols not available in Literary Braille. The Nemeth Braille Code is not an expanded version of the Literary Braille Code [10].

C- Computer Braille Code (CBC)

Computer Braille Code was developed in order to transcribe materials relating to computers. It uses some symbols from Literary Braille, some from Nemeth Code, and some symbols of its own. Computer Braille Code requires file names, the exact count of blank spaces, and other details that are required for computer programming and the use of computers [10].

D- Music Braille Code

Music Braille Code is a completely different Braille code. An extensive knowledge of music is necessary for the accurate transcription of music into Braille. [10].

2.2 Braille Letters and Characters:

In this paper, the literary grade I Braille code will be considered. The first ten letters of the alphabet and the numbers 1 through 0 are formed using only the top four dots (1, 2, 4, and 5). Adding dot 3 forms the next ten letters, and adding dot 6 forms the last five letters (except w) and the words: (and, for, of, the, and with). Omitting dot 3 from the letters U-Z and the five word symbols forms nine digraphs (ch, gh, sh, th, wh, ed, er, ou, and ow) and the letter w [1]. The numbers (1, 2... 0) are announced by a sign using dots 3, 4, 5, and 6 before the letters (A, B... J). The use of dot 6 just before a letter indicates a capital letter as shown in the Figure (3) [1] [12].

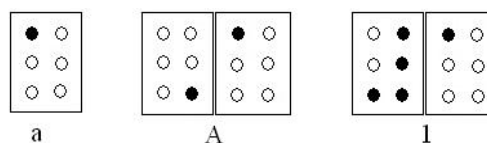


Figure (3): The use of two cells to express capital letters and numbers.

3- Neural Networks

An Artificial Neural Network (ANN), often just called a "Neural Network" (NN), is a mathematical model or computational model composed of simple elements operating in parallel. These elements are inspired by biological nervous systems. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows

through the network during the learning phase.

In more practical terms neural networks are non-linear statistical data modeling tools. They can be used to model complex relationships between inputs and outputs or to find patterns in data. [13] [14].

The universal approximation theorem for neural networks states that: "every continuous function that maps intervals of real numbers to some output interval of real numbers can be approximated arbitrarily closely by a multi-layer perceptron with just one hidden layer". This result holds only for restricted classes of activation functions, e.g. for the sigmoidal functions.

4- Design of English to Braille converter using Neural Networks

Using MATLAB programming, a neural network will be designed to recognize and to convert English

characters into Braille characters. The neural network should be able to classify noisy characters as well as noise free characters.

A page that is scanned by an imaging device is converted into matrices of 400 element columns that represent a 20*20 grid for each character. The matrix should have the values of 1's to represent black, and 0's elsewhere. Sometimes, the imaging device is not perfect and the letters may suffer from noise. Here the neural network should be able to recognize the true characters and generate an output according to each character. However, the method of scanning and converting the image into matrices is out of the scope of this paper.

The output of the network is used to generate the corresponding Braille character according to the Braille rules. The generated Braille characters are saved as a matrix of 2*6 elements for each character.

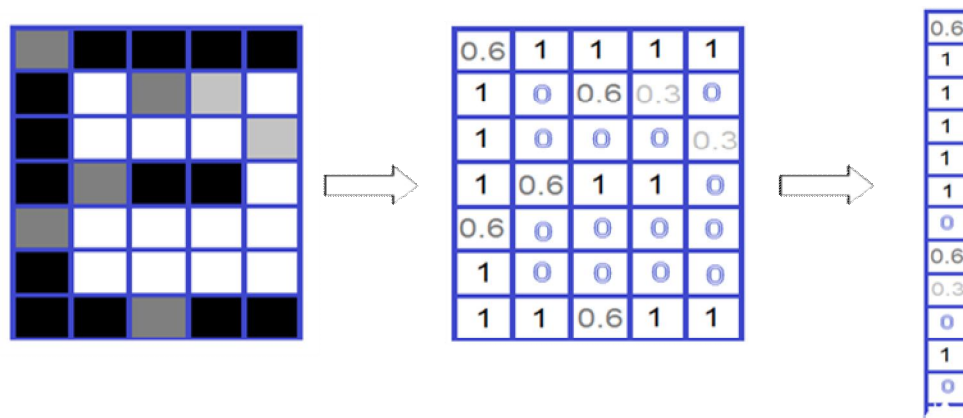


Figure (4): Converting noisy E character into 35 element array.

As in English representation; each Braille dot is represented by 1 in its corresponding place in the output matrix, and zero elsewhere. In this paper, the artificial neural network will be designed and trained to recognize 63 characters :(52 English capital and small letters, 10 numbers and the space) using back-propagation training algorithm. The network is trained using the patterns of the input matrix (English_char). This matrix has a size of 400*63 elements; each column of this matrix contains the data of a perfect character. Capital letters (A-Z) are placed in the columns (1-26) and then the small letters, followed by numbers and the (space). The output of the network is represented by the identity matrix (Targets) that has 63*63 elements. The steps of the algorithm are shown in the Figure (5). The neural network needs 400 inputs in the input layer and 63 outputs in its output layer. The output of the network should be 1 in the position of the detected character and 0 elsewhere. For example; if character (B) is the input then the output should have 1 in the 2nd element in the output vector and 0 elsewhere. Finally, the MATLAB program is used to generate Braille characters according to the output of the neural network.

4.1- Initialization of the Data Characters:

In order to design and train the neural network, data characters should be initialized and the following steps should be made:

- Load Perfect character images.

- Resize and reshape the characters into a column matrix.
- Arrange characters' columns side by side in the matrix (English_char).
- Create each Braille character as a 6*2 array.
- Arrange the arrays of the Braille characters in the matrix (braille) so that it contains all of the Braille characters (side by side).
- Create the matrices (capital_next) and (number_next) as 6*2 for each.
- Create the matrix (Targets) which is the identity matrix of 63*63.

4.2- Creating the Neural Network:

In order to create and train the Neural Network, a feed-forward one is created with 400 inputs in the input layer, a different number of hidden layers and 63 outputs in its output layer. Sigmoid activation functions are used in hidden layers and bipolar linear activation functions are used in the nodes of output layer. The resilient back-propagation algorithm (trainrp) is used to train this Neural Network. This method supports training with validation and test vectors.

4.3- Training the Neural Network:

1- Training without noise:

First, the network was trained to recognize the noise free characters with the learning rate of 10^{-3} and an error goal of 10^{-6} .

2- Training with noise:

Second, after updating the weights to recognize the ideal inputs with a very small error, the network was forced to recognize noisy input characters. The noise is created by generating random numbers with a mean of 0 and standard deviation (std) of 0.2. This learning was repeated 35 times; for each time it was trained with 14 copies of the input; half of these copies have a noise of std 0.2 added to them. More error was expected because more input vectors (some with noise) are used; therefore, the error goal was reduced to 10^{-3} .

3- Training without noise:

Third, to make sure that the network always responds correctly to the ideal inputs, the network was trained once more with ideal inputs. The mean square error goal was decreased to 10^{-3} .

5. Simulation of the Neural Network as a Converter:

To simulate the variable noise from the scanner; a random noise was added to the input signal with std of 0.05 and the detected English characters as well as the corresponding Braille characters were drawn as in Figure (6).

Using a noise of 0.1 std resulted in no mistakes during the detection and conversion, shown in Figure (7). Increasing the noise into 0.2 resulted in (0-2) mistakes for each trial as shown in Figure (8).

Conclusions:

It can be concluded that:

A feed forward artificial neural network was designed and tested to convert English characters into grade I literary Braille code, taking into account the nonlinearity of the conversion because of Braille rules. The network was designed with minimum structure including a number of layers, and a number of neurons in each layer. Noisy input patterns were introduced to the network including the addition of a noise of std 0.2 to all characters which resulted in one or two characters to be detected wrong for each time running the program. A pattern of characters without and with noise were tested using the designed N.N and the results were satisfying.

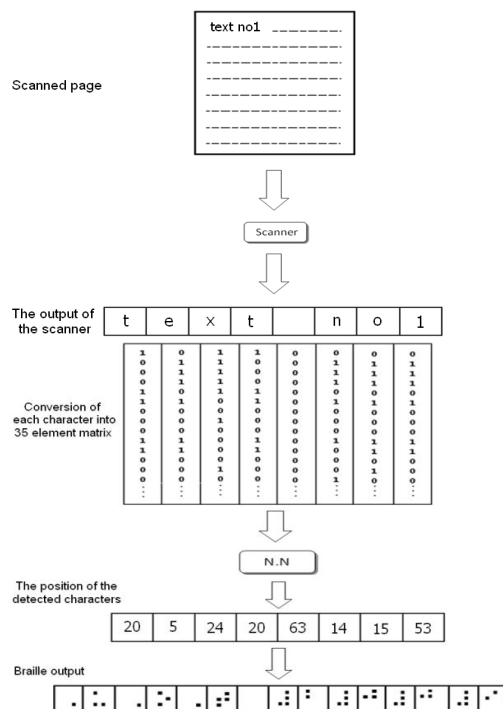


Figure (5): conversion steps of English characters into Braille.

After several trials, using the following steps to get minimum structure of the designed network, it was found that the best structure is achieved with 190 neurons in the hidden layer.

abCdEf gHi jkl m nopqr St UVWXYZ ABCDE
FGHI JKLMNOPQRS TUVWXYZ 1 2 3 4 5 6 7 8 9 0

Figure (6-a): Noisy input characters with 5% noise.

abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
FGHIJKLMNOPQRSTUVWXYZ1234567890

Figure (6-b): Detected characters using the N.N.

```
***** APPENDIX A *****  
A  
B  
C  
D  
E  
F  
G  
H  
I  
J  
K  
L  
M  
N  
O  
P  
Q  
R  
S  
T  
U  
V  
W  
X  
Y  
Z
```

Figure (6-c): Corresponding output Braille characters

abCdEf gHi jkl m nopqrStUvwXyZABCDE
FGHI JKLMNOPQRSTUvwXyZ1234567890

Figure (7-a): Noisy input characters with 10% noise.

abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
FGHIJKLMNOPQRSTUVWXYZ1234567890

Figure (7-b): Detected characters using the N.N.

1. **RESEARCH DESIGN**
 2. **RESEARCH DESIGN**
 3. **RESEARCH DESIGN**
 4. **RESEARCH DESIGN**
 5. **RESEARCH DESIGN**
 6. **RESEARCH DESIGN**
 7. **RESEARCH DESIGN**
 8. **RESEARCH DESIGN**
 9. **RESEARCH DESIGN**
 10. **RESEARCH DESIGN**
 11. **RESEARCH DESIGN**
 12. **RESEARCH DESIGN**
 13. **RESEARCH DESIGN**
 14. **RESEARCH DESIGN**
 15. **RESEARCH DESIGN**
 16. **RESEARCH DESIGN**
 17. **RESEARCH DESIGN**
 18. **RESEARCH DESIGN**
 19. **RESEARCH DESIGN**
 20. **RESEARCH DESIGN**
 21. **RESEARCH DESIGN**
 22. **RESEARCH DESIGN**
 23. **RESEARCH DESIGN**
 24. **RESEARCH DESIGN**
 25. **RESEARCH DESIGN**
 26. **RESEARCH DESIGN**
 27. **RESEARCH DESIGN**
 28. **RESEARCH DESIGN**
 29. **RESEARCH DESIGN**
 30. **RESEARCH DESIGN**
 31. **RESEARCH DESIGN**
 32. **RESEARCH DESIGN**
 33. **RESEARCH DESIGN**
 34. **RESEARCH DESIGN**
 35. **RESEARCH DESIGN**
 36. **RESEARCH DESIGN**
 37. **RESEARCH DESIGN**
 38. **RESEARCH DESIGN**
 39. **RESEARCH DESIGN**
 40. **RESEARCH DESIGN**
 41. **RESEARCH DESIGN**
 42. **RESEARCH DESIGN**
 43. **RESEARCH DESIGN**
 44. **RESEARCH DESIGN**
 45. **RESEARCH DESIGN**
 46. **RESEARCH DESIGN**
 47. **RESEARCH DESIGN**
 48. **RESEARCH DESIGN**
 49. **RESEARCH DESIGN**
 50. **RESEARCH DESIGN**
 51. **RESEARCH DESIGN**
 52. **RESEARCH DESIGN**
 53. **RESEARCH DESIGN**
 54. **RESEARCH DESIGN**
 55. **RESEARCH DESIGN**
 56. **RESEARCH DESIGN**
 57. **RESEARCH DESIGN**
 58. **RESEARCH DESIGN**
 59. **RESEARCH DESIGN**
 60. **RESEARCH DESIGN**
 61. **RESEARCH DESIGN**
 62. **RESEARCH DESIGN**
 63. **RESEARCH DESIGN**
 64. **RESEARCH DESIGN**
 65. **RESEARCH DESIGN**
 66. **RESEARCH DESIGN**
 67. **RESEARCH DESIGN**
 68. **RESEARCH DESIGN**
 69. **RESEARCH DESIGN**
 70. **RESEARCH DESIGN**
 71. **RESEARCH DESIGN**
 72. **RESEARCH DESIGN**
 73. **RESEARCH DESIGN**
 74. **RESEARCH DESIGN**
 75. **RESEARCH DESIGN**
 76. **RESEARCH DESIGN**
 77. **RESEARCH DESIGN**
 78. **RESEARCH DESIGN**
 79. **RESEARCH DESIGN**
 80. **RESEARCH DESIGN**
 81. **RESEARCH DESIGN**
 82. **RESEARCH DESIGN**
 83. **RESEARCH DESIGN**
 84. **RESEARCH DESIGN**
 85. **RESEARCH DESIGN**
 86. **RESEARCH DESIGN**
 87. **RESEARCH DESIGN**
 88. **RESEARCH DESIGN**
 89. **RESEARCH DESIGN**
 90. **RESEARCH DESIGN**
 91. **RESEARCH DESIGN**
 92. **RESEARCH DESIGN**
 93. **RESEARCH DESIGN**
 94. **RESEARCH DESIGN**
 95. **RESEARCH DESIGN**
 96. **RESEARCH DESIGN**
 97. **RESEARCH DESIGN**
 98. **RESEARCH DESIGN**
 99. **RESEARCH DESIGN**
 100. **RESEARCH DESIGN**

Figure (7-c): Corresponding output Braille characters.

abcdefghijklmnopqrstuvwxyz ABCDE
FGHIJKLMNOPQRSTUVWXYZ 1234567890

Figure (8-a): Noisy input characters with 20% noise.

abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
FGHIJKLMNOPQRSTUVWXYZ1234567890

Figure (8-b): Detected characters using the N.N.

1. **Identify the main components of the system.**

Figure (8-c): Corresponding output Braille characters.

Finally, it can be concluded that the conversion of English to Braille using Neural Network is a fast and accurate method. It can be assumed as a general method for any script written in English language.

References:

- [1] Wikipedia the free encyclopedia, 2008 en.wikipedia.org/wiki/Braille
- [2] Avi-Itzhak H. I., Diep T. A., and Garland H., "High Accuracy Optical Character Recognition Using Neural Networks with Centroid Dithering", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 17, No. 2, PP. 218-224, Feb 1995.
- [3] Oh S., Suen C. Y., "Distance features for neural network-based recognition of handwritten characters", International Journal on Document Analysis and Recognition (IJ DAR), c Springer-Verlag, Vol. 1, PP. 73-88, 1998.
- [4] Garriss M. D., Wilson C. L., and Blue J. L., Aug. "Neural Network-Based Systems for Handprint OCR Applications", IEEE Transaction on Image Processing, Vol. 7, No. 8, PP. 1097-1112, 1998.
- [5] Khatatneh K., El Emary I. M. M. and Al- Rifai B., "Probabilistic Artificial Neural Network For Recognizing the Arabic Hand Written Characters", Journal of Computer Science, Vol. 3, No. 12, PP. 881-886, 2006,
- [6] Shilbayeh N. F. and Iskandarani M. Z., "Effect of Hidden Layer Neurons on the Classification of Optical Character Recognition Typed Arabic Numerals", Journal of Computer Science, Vol. 4, No. 7, PP. 578-584, 2008.
- [7] Sriskanthan N. and Subramanian K. R., "Braille Display Terminal for Personal Computers", IEEE Transactions on Consumer Electronics, Vol. 36, No. 2, PP. 121-128, May 1990.
- [8] Blenkhorn P., June 1995, "A System for Converting Braille into Print", IEEE Transaction on Rehabilitation Engineering, Vol. 3, No. 2, 215-221.
- [9] Karmakar S., Chatterjee R. P. and Dutta U., "Improvement in quality testing of Braille printer output with Euclidean distance measurement using camera calibration", International Journal of Engineering, Science and Technology Vol. 2, No. 1, pp. 35-48, 2010,.
- [10] Alternative Method Research www.braillefont.com/index.html
- [11] MK Design, 2004, "The Professional Sign Consultancy Service". www.mkdesign.free-online.co.uk/Braille.htm
- [12] www.cnib.ca/en/living/braille/braille
- [13] Peter A., Burgsteiner H., Maass W., "A learning rule for very simple universal approximators consisting of a single layer of perceptrons", Neural Networks, Vol. 21, No. 5, PP. 786-795, 2008.
- [14] Zurada J. M., Introduction to Artificial Neural Networks, 2nd Edition, India, 1996.