Design of FPGA Based Handwritten Character Recognition System using Feed forward Neural Network

Mr. Chirantar Wanjari
PG Student, Electronics Engineering Department, G. H. Raisoni College of Engineering, Nagpur

Prof. P.M. Ghutke
Electronics Engineering Department,
G. H. Raisoni College of Engineering, Nagpur

Abstract: This paper presents the FPGA based handwritten character recognition system for any characters. This recognition system algorithm is based on the feed forward neural Network which includes the learning and classification .this system is designed on the Xilinx13.1, Sparten-3 FPGA. For this system, the some database is created on personal computer. The neural network operates on these databases which are in text format. This designed recognition system is tested and achieved recognition rate up to 99%.

Keywords- Character recognition, Recognition rate, feed forward neural network.

I. INTRODUCTION: -

Some handwritten character recognition system used SOM neural network which has less recognition rate up to 86% [7]. Data in such systems is generally collected by writing on a passive medium such as paper and then the system scans or photographs the inputs. Man-machine communication is based on the use of keyboard. This interface is not well suited to users who have not mastered the keyboard. Therefore, automatic handwriting recognition systems have been developed. This system proposed implementation of handwritten recognition using back propagation neural network with the help of VHDL coding on the FPGA. FPGA implementation provides modular design for simplicity. Feed forward neural network includes the two phases like learning and classification. This neural network provides average recognition rate up to 99%.

II. SCHEME OF RECOGNITION SYSTEM

This scheme includes the steps of designing recognition system as shown in fig 1. Initially, drawn the pattern on the paint tool on the PC. Then converted that pattern into the 128*128 Pixel size. After that conversion, this 128*128 Pixel size converted into text format. Then this

128*128 text size is divided into 16 number of 32*32 size region for processing. This 32*32 size region fed as input to the feature extraction module.

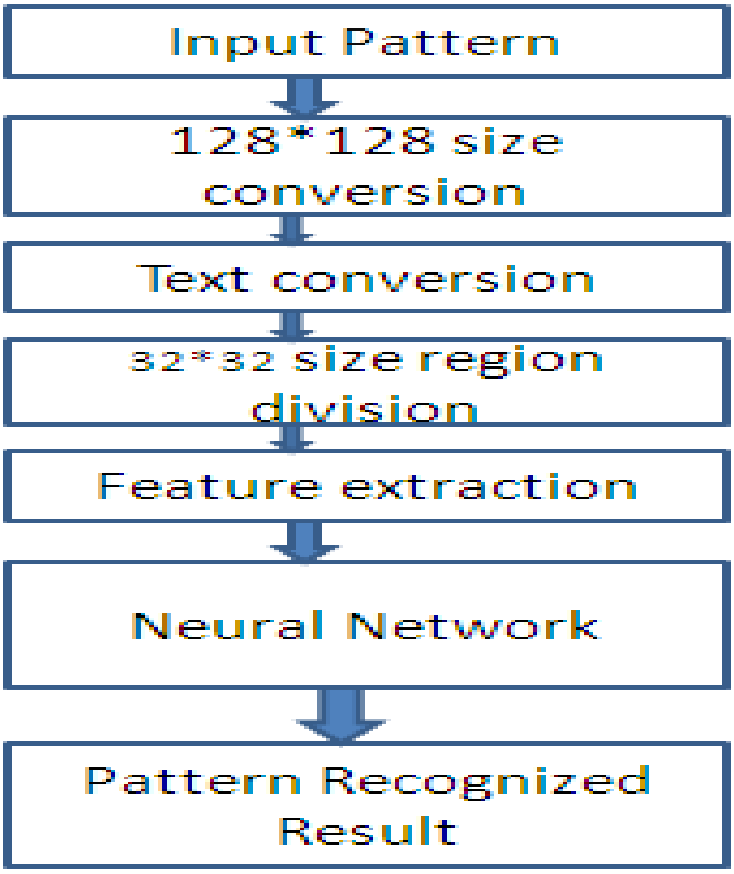

Fig 1. Scheme of Recognition System

Then extracted features from feature extraction model fed as input to neural network. Finally, neural network shown the pattern recognized result through simulation result.

III. FEATURE EXTRACTION MODULE:

Each 32*32 size region fed as input to the feature extraction model. One region is loaded at a time to extract the density features 1–16. The remaining features 17–44 are derived from features 1–16. The starting address of the regions is internally stored inside the feature extraction module. The module reads the 32 rows of each region through its memory interface and buffers them. The initial region address is set, memory transfer signals are asserted, and the data is transferred and buffered. The address is adjusted to point to the next row within a region by increments of 32 bytes.

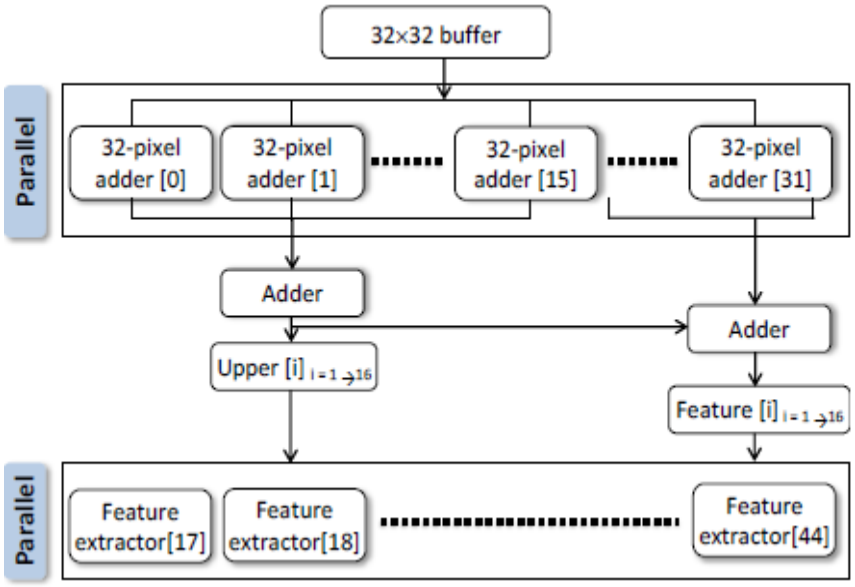

Fig.2. Feature extraction module

After the end of one region transfer, the pixels of each row are summed up in parallel using 32 multi-input adders. Then the 32 partial sums are aggregated using another multi-input addition stage. This stage generates two results for each Region *i*: the sum of Rows 0–15 (*Upper[i]*) and the sum of Rows 0–31 (*Feature[i]*). These two results are buffered. The next region address is then loaded and the same computation is repeated for the remaining 15 regions. Features 17–44 are computed in parallel from the buffered regions results. Each of the Features 33–40 is computed by adding the corresponding two horizontally-adjacent features of the range 1–16. Each of the Features 40–44 is computed by adding the corresponding two horizontally-adjacent features of the range 33–40. The odd-numbered features in the range 17–32 are found by summing two horizontally-adjacent *Upper* results; whereas the even-numbered features are found by subtracting the odd-numbered feature from the corresponding feature in the range 33–40. Finally, all features are saved in the features and weights memory.

IV.NEURAL NETWORK RECOGNITION MODULE:

Features extracted from the feature extraction module fed as input to the neural network recognition module through feature and weight memory. All arithmetic computations within this neuron function are performed in parallel. The multiplications of the neuron inputs and their weights in addition to the multi input addition are performed in parallel as shown in Fig.3.

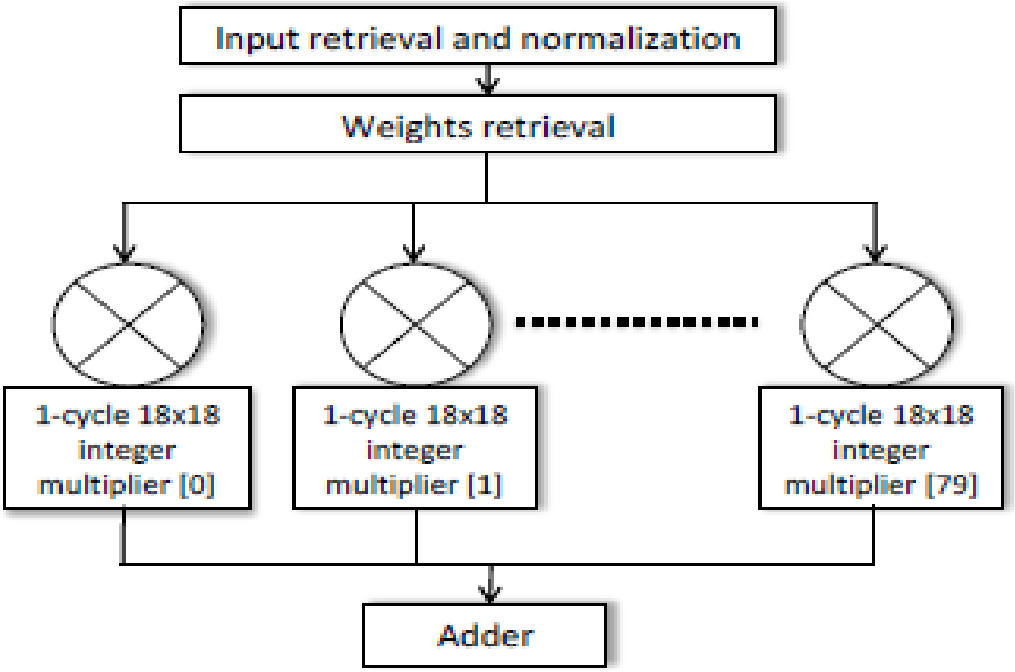

Fig.3.Neural network recognition module

The neuron function performs only 44 parallel multiplications for the hidden layer neurons and performs 80 parallel multiplications for the output layer neurons. The embedded FPGA multipliers are utilized for input normalization and weight multiplication. The input normalization is only performed on the inputs of the hidden layer.

V. DESIGN OF RECOGNITION SYSTEM:

The whole recognition system consisting of feature extraction module and neural network recognition module has designed using VHDL coding on the Xilinx13.1 and verified the result for recognition rate.

VI. RESULT

The 1000 samples of each English letters is collected. These samples of letters fed to the recognition system. Then it is cleared that, this system has average recognition rate up to 99%.

The simulation result is as shown in fig.4

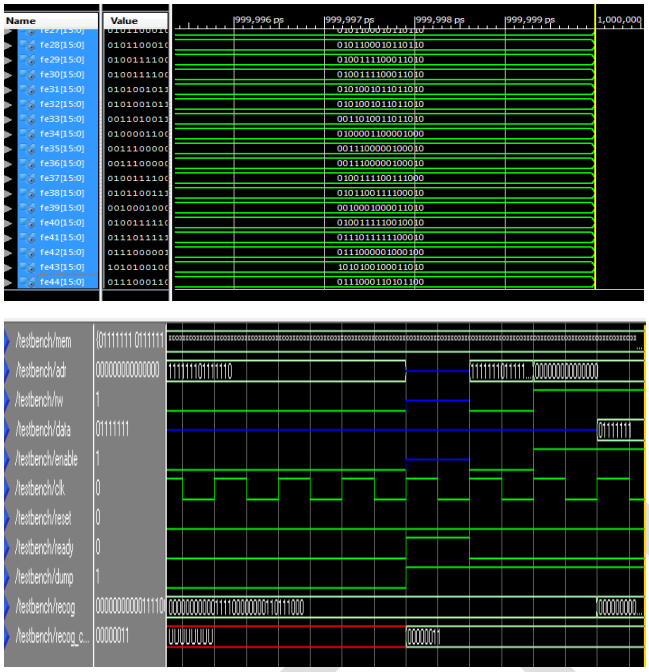

Fig.4 Simulation result

VII. FPGA SYNTHESIS REPORT:

Logic Utilization	Used	Available	Utilization
Total Number Slice Registers	936	9,312	10%
Number used as Flip Flops	879		
Number used as Latches	57		
Number of 4 input LUTs	2,862	9,312	30%
Number of occupied Slices	1,742	4,656	37%
Number of Slices containing only related logic	1,742	1,742	100%
Total Number of 4 input LUTs	3,238	9,312	34%
Number of bonded IOBs	40	232	17%
Number of RAMB16s	3	20	15%
Number of BUFGMUXs	9	24	37%
Number of MULT18X18SIOs	1	20	5%

VIII. EXPERIMENTAL RESULT:

Every one of the 100 peoples writes numeral character ‘A’-‘J’ for ten times, then let computer

Recognize it, the result is Shown in following table:

English character	No.of samples	Recognition rate
A	1000	99.26%
B	1000	99.17%
C	1000	99.10%
D	1000	99.01%
E	1000	98.90%
F	1000	98.50%
G	1000	99.05%
H	1000	99.15%
I	1000	99.20%
J	1000	99.47%

IX. CONCLUSION:

This paper presented the design of FPGA based handwritten English character recognition system using back propagation neural network. Back propagation neural network was used to categorize and identify the English characters written by user. This neural network can also be used for other language characters.

REFERENCES:

[1] M. Moradi, M. A. Pourmina and F. Razzazi, "A new method of FPGA Implementation of Farsi Handwritten Digit Recognition", European Journal of Scientific Research, vol. 39, February 2010, pp. 309-315

[2] R. K. Sadykhov, L. P. Podenok, V. A. Samokhval and A. A. Uvarov, "The System for Handwritten Symbol and Signature Recognition using FPGA Computing", Lecture Notes in Computer Science, November 2004.

[3] S. Knerr, L. Personnaz, and G. Dreyfus, "Handwritten digit recognition by neural networks with single-layer training", IEEE Trans. on Neural Networks, *vol. 3, June 1992, pp. 962-968.*

[4] D. Gil and M. Johnsson, "Supervised SOM Based Architecture versus Multilayer Perceptron and RBF Networks", Proc. of the .Linköping Electronic Conf., pp. 15 – 24, 2010.

[5] S. Arora, D. Bhattacharjee, Mita Nasipuri, L. Malik , M. Kundu and D. K. Basu, "Performance Comparison of SVM and ANN for Handwritten Devnagari Character Recognition", IJCSI Int. Journal of Computer Science Issues, vol. 7, June 2010, pp. 18-26.

[6] B. K. Verma, "Handwritten Hindi character recognition using multilayer perceptron and radial basis function neural networks", Proc. of the IEEE Int. Conf. on Neural Networks, pp. 2111-2115, 2002.

[7] ZILI-CHEN, ZUXUE-WEI, " A Handwriting Numeral Character Recognition System", ©2010 IEEE.

[8] Emmanuel Bouvett, Owen Casha," An FPGA Embedded System Architecture for Handwritten Symbol Recognition", ©2012 IEEE.

[9] Ashraf Suyyagh, Gheith Abandah," FPGA Parallel Recognition Engine for Handwritten Arabic Words",2013 IEEE

[10]K.S. Sharma, A.R. Karwankar, and A.S. Bhalchandra, "Devnagari character recognition using self organizing maps", Proc. of the IEEE Int. Conf. on Communication Control and Computing Tech., pp. 687-691, 2010.

[11] S. Shelke and S. Apte, "Multistage Handwritten Marathi Compound Character Recognition Using Neural Networks", Journal of Pattern Recognition Research 2, 2011, pp. 253-268.

[12] XILINX Corp., "PicoBlaze 8-bit Embedded Microcontroller User Guide", 2010.

[13] T. Kohonen, "Self-organizing maps", 3rd edition, Springer-Verlag New York, 2001.

[14] M. Cutajar, E. Gatt, J. Micallef, I. Grech and O. Casha: "Digital Hardware Implementation of Self-Organising Maps", 15th IEEE Melecon, Valletta, Malta, pp. 1123-1128, April 2010.