ISSN No. 0976-5697
Volume 2, No. 6, Nov-Dec 2011
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

Segmentation of Textures Using Echo State Neural Network

Shobarani*
Research Scholar,
Department of Computer Science,
Mother Teresa Women's University,
Kodaikanal, India
rshobarani@gmail.com

Dr. S. Purushothaman
Dean,
Udaya School of Engineering, Vellamodi,
Ammandivilai, India

Abstract: This paper presents texture segmentation concept using Echo state neural network and Fuzzy Logic. The data set used is the textile textures. The image is split into 3 X 3 windows. The features of the windows are presented to the input layer of the echo state network. The target for each window is assigned. The number of reservoirs in the network decides the segmentation performance.

Keywords: image segmentation, Echo State Neural network, unsupervised method, fuzzy membership function and textures.

I. INTRODUCTION

The problem of classifying materials from their imaging, without imposing any constraints on, or requiring any *a priori* knowledge of, the viewing or illumination conditions under which these images were obtained. The classification problem is for given a single, uncalibrated textured image[2-5], classify it into one of a set of pre-learnt classes. Classifying textures from single image is a very demanding task. What makes the problem so hard is that unlike other forms of classification, where the objects being categorized have a definite structure which can be captured and represented, most textures have large stochastic variations which make them difficult to model. Textured materials often undergo change in their imaged appearance with variations in illumination and camera pose. Dealing with this successfully is one of the main tasks of any classification algorithm. Two materials when photographed under very different imaging conditions can appear to be quite similar. In this research work, the investigations carried out on textile textures and mosaic using fuzzy logic and Echo State Neural Network for improved texture segmentation. Texture is the result of spatial variation in pixel intensities in an image. Texture is useful in a variety of applications and has been a subject of intense study by many researchers. Recognition of homogeneous image regions using texture properties is an important application. The goal is to do classification of input image with the existing textures template. Texture segmentation is achieved using texture properties.

II. PREVIOUS WORK

Yining Deng et. al., (2001) proposed a class map method for texture segmentation. Lin Ma et. al., (2009) proposes Ant Colony Optimization (ACO) for texture segmentation. Mohand Saïd et. al., (2007) propose an automatic segmentation of color-texture images with arbitrary numbers of regions. The approach combines region and boundary information and uses active contours to build a partition of the image. Wong W. K. et. al., (2009), combines wavelet transform with the back propagation (BP) neural network. The smooth sub-image at a certain resolution level using the pyramid wavelet transform was obtained. The study uses the direct thresholding method, which is based on wavelet transform smooth sub-images from the use of a quadrant mean filtering method, to attenuate the texture background and preserve the anomalies. The images are then segmented by threshold processing and noise filtering. Nine characteristic variables based on the spectral measure of the binary images were collected and input into a BP neural network to classify the sample images. Shoudong Han et. al., (2011) proposes a novel texture segmentation approach using independent-scale component-wise Riemannian-covariance Gaussian mixture model (ICRGMM) in Kullback-Leibler (KL) measure based multi-scale nonlinear structure tensor (MSNST) space.

III. SCOPE OF PRESENT WORK

In this work, a systematic approach has been developed to extract textures from the given texture images. The features are extracted using Echo State Neural Network. The extracted features are used to segment the image using supervised method, fuzzy logic and Echo State Neural Network. The major combinations of the algorithms developed are as follows:

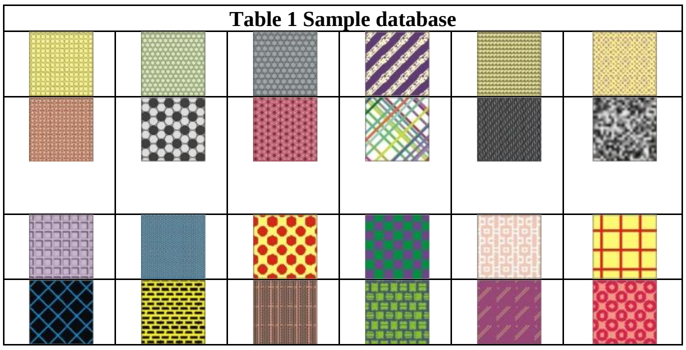
a. Echo State Neural Network using recurrent concept.
b. Fuzzy logic using supervised method.

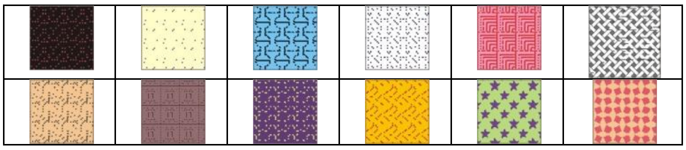
IV. SIMULATION OF DATA

A. *Data Collection:*

A sample database has been presented in Table 1. The table presents few textile textures.

Table 1 Sample database





V. MATERIALS AND METHODS

A. *Materials:*

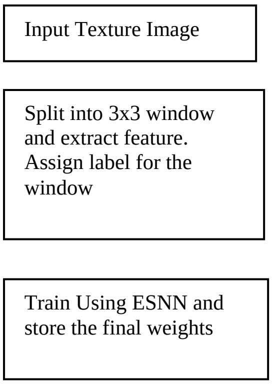


Figure 1
Training the ESNN

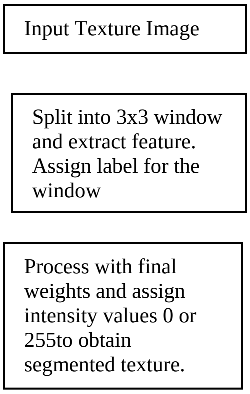


Figure 2
Segmenting images Using ESNN

Step 1: split the image into 3 X 3 window (overlapping).
Step 2: Train the ESNN by initializing the random weights. At the end of training store the final weights.
Step 3: input sample texture image and process with the final weights to obtain the segmented texture image.

B. *Methods:*

a. *Echo State Neural Network:*

An Artificial Neural Network (ANN) is an abstract stimulation of a real nervous system that contains a collection of neuron units, communicating with each other via axon connections. Artificial neural networks are computing elements which are based on the structure and function of the biological neurons. These networks have nodes or neurons which are described by difference or differential equations.

Dynamic computational models require the ability to store and access the time history of their inputs and outputs. The most common dynamic neural architecture is the Time-Delay Neural Network (TDNN) that couples delay lines with a nonlinear static architecture where all the parameters (weights) are adapted with the back propagation algorithm. Recurrent Neural Networks (RNNs) implement a different type of embedding that is largely unexplored. RNNs are perhaps the most biologically plausible of the Artificial Neural Network (ANN) models. One of the main practical problems with RNNs is the difficulty to adapt the system weights. Various algorithms, such as back propagation through time and real-time recurrent learning, have been proposed to train RNNs; however, these algorithms suffer from computational complexity, resulting in slow training, complex performance surfaces, the possibility of instability, and the decay of gradients through the topology and time.

The problem of decaying gradients has been addressed with special processing elements (PEs). The ESNN, Figure 5, with a concept new topology has been found by ESNN possesses a highly interconnected and recurrent topology of nonlinear PEs that 12 constitutes a reservoir of rich dynamics and contains information about the history of input and output patterns. The outputs of this internal PEs (echo states) are fed to a memory less but adaptive readout network (generally linear) that produces the network output.

The interesting property of ESNN is that only the memory less readout is trained, whereas the recurrent topology has fixed connection weights. This reduces the complexity of RNN training to simple linear regression while preserving a recurrent topology, but obviously places important constraints in the overall architecture that have not yet been fully studied.

The echo state condition is defined in terms of the spectral radius (the largest among the absolute values of the Eigen values of a matrix, denoted by (|| ||) of the reservoir's weight matrix (|| W || < 1). This condition states that the dynamics of the ESNN is uniquely controlled by the input, and the effect of the initial states vanishes. The current design of ESNN parameters relies on the selection of spectral radius. There are many possible weight matrices with the same spectral radius, and unfortunately they do not perform at the same level of mean square error (MSE) for functional approximation. The recurrent network is a reservoir of highly interconnected dynamical components, states of which are called echo states. The memory less linear readout is trained to produce the output. Consider the recurrent discrete-time neural 13 network given in Figure 6 with M input units, N internal PEs, and L output units.

The value of the input unit at time n is u(n) = [u1(n), u2(n), . . . , uM(n)]T ,

The internal units are x(n) = [x1(n), x2(n), . . . , xN(n)]T , and

Output units are y(n) = [y1(n), y2(n), . . . , yL (n)]T.

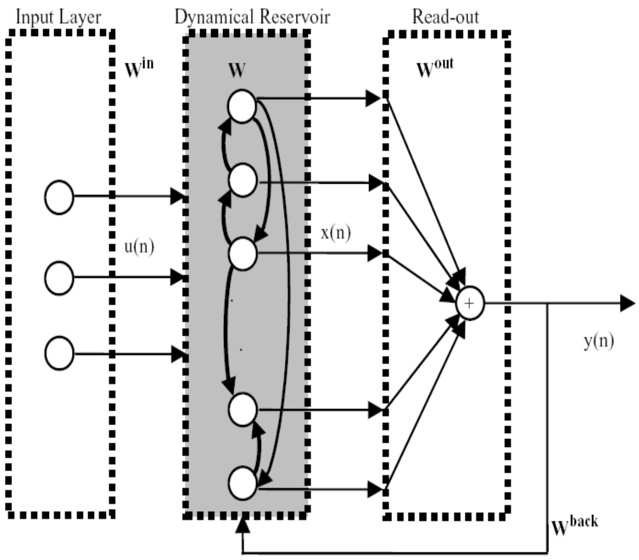


Figure 3 An Echo State Network (ESNN)

The connection weights are given

- in an (N x M) weight matrix $W^{back} = W_{ij}^{back}$ for connections between the input and the internal PEs,
- in an N × N matrix $W^{in} = W_{ij}^{in}$ for connections between the internal PEs
- in an L × N matrix $W^{out} = W_{ij}^{out}$ for connections from PEs to the output units and

• in an N × L matrix $W^{back} = W_{ij}^{back}$ for the connections that project back from the output to the internal PEs. The activation of the internal PEs (echo state) is updated according to

x(n + 1) = f(W^{in} u(n + 1) + Wx(n) + W^{back} y(n)), where f = (f1, f2, . . . , fN) are the internal PEs' activation functions. Here, all fi's are hyperbolic tangent functions

$$\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$$

The output from the readout network is computed according to y(n + 1) = fout(Woutx(n + 1)), where

$f^{out} = (f_1^{out}, f_2^{out} ...... f_L^{out})$ are the output unit's nonlinear functions. Generally, the readout is linear so f^{out} is identity (purushothaman 2008).

The algorithm for training the ESNN is as follows:

Step 1: Read a Pattern (I) (texture image feature) and its Target (T) value.
Step 2: Decide the number of reservoirs.
Step 3: Decide the number of sides in the input layer = length of pattern.
Step 4: Decide the number of sides in the output layer = number of target values.
Step 5: Initialize random weights between input and hidden layer (Ih) hidden and output .
Step 6: Calculate F=Ih*I.
Step 7: Calculate TH = Ho * T.
Step 8: Calculate TT = R*S.
Step 9: Calculate S = tan h(F+TT+TH).
Step 10: Calculate a = Pseudo inverse (S).
Step 11: Calculate Wout = a * T and store Wout for testing.

The algorithm for testing the ESNN is as follows:

Step 1: Read a Pattern (I) (texture image feature).
Step 2: Calculate F=Ih*I.
Step 3: TH = Ho * T.
Step 4: TT = R*S.
Step 5: S = tan h(F+TT+TH).
Step 6: a = Pseudo inverse (S).
Step 7: estimated = a * Wout
Step 8: Classify it

Figure 4 shows the error between estimated and target values. The curve oscillates and minimum is obtained at 22 nodes.

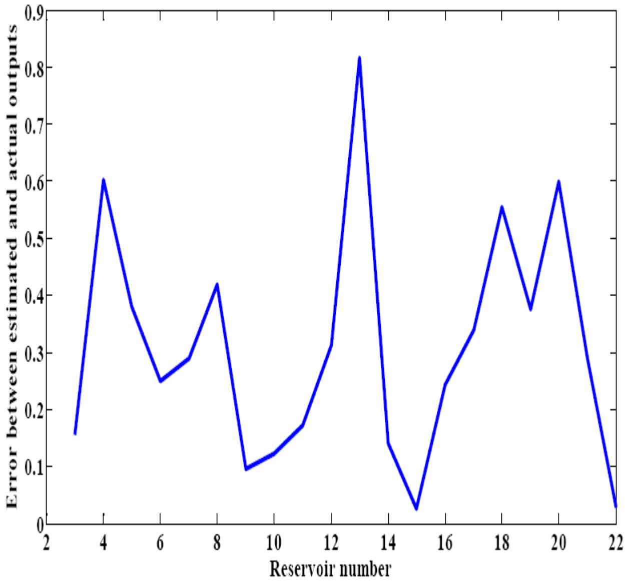


Figure 4: Error between estimated and actual output

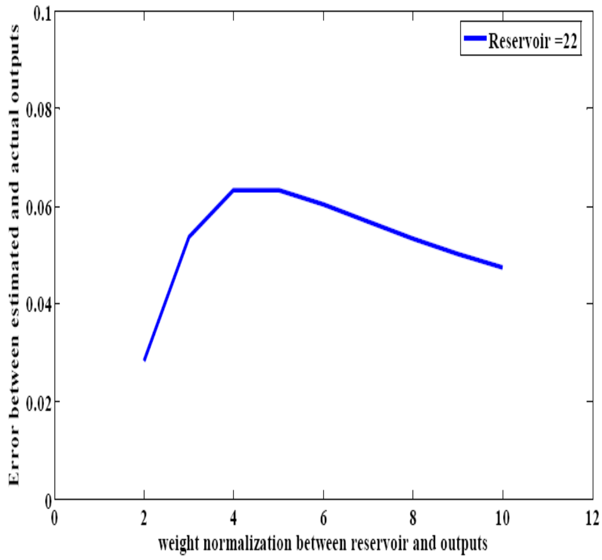


Figure 5: Error between estimated and actual output

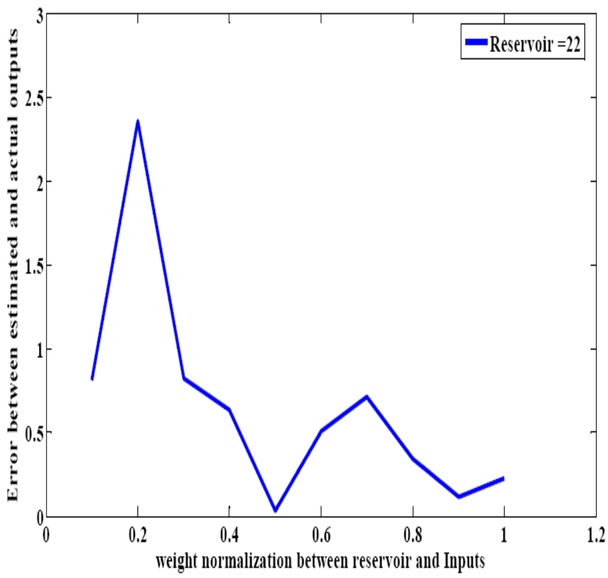


Figure 6: Error between estimated and actual output

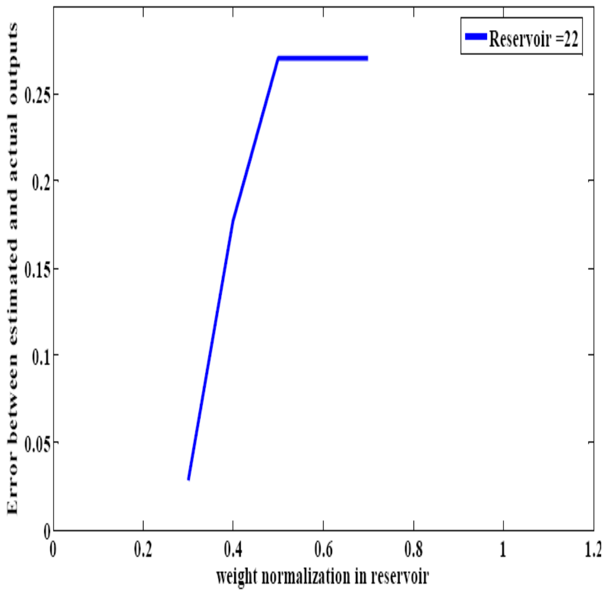
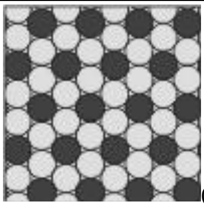
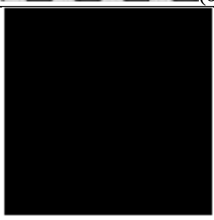
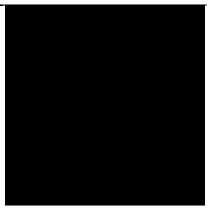
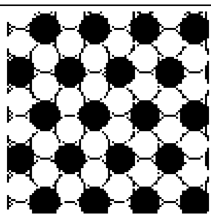
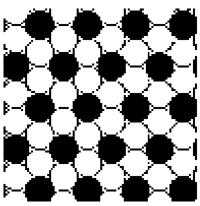


Figure 7: Error between estimated and actual output

Table 2 Echo state network segmentation output	
Reservoir	 (original texture)
4	 (segmented)
5	 (segmented)
6	 (segmented)
7	 (segmented)

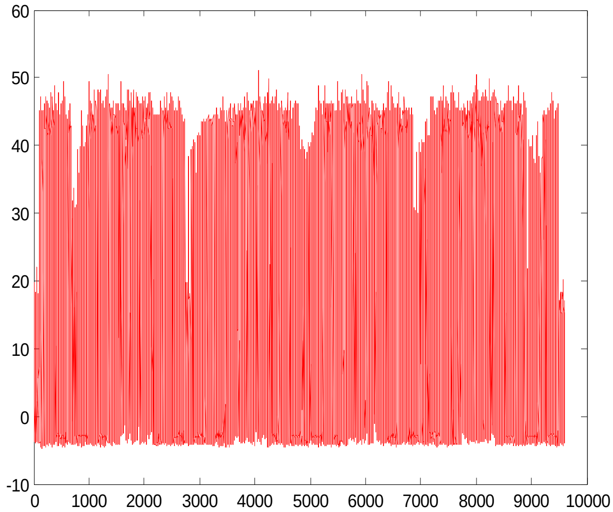


Figure 8: Echo state network output

VI. RESULTS AND DISCUSSIONS

Figure 4 shows the error between estimated and target values. The curve oscillates and minimum is obtained at 22 nodes. In Figure 5, the change of weight values and their impact in estimation of ESNN is presented. The error increases and decreases. The x axis represents the change in the weight values in output and hidden layer. In Figure 6, the change of weight values and their impact in estimation of ESNN is presented. The error increases and decreases continuously. The x axis represents the change in the weight values in input and hidden layer. In Figure 7, the change of weight values and their impact in estimation of ESNN is presented. The error increases and decreases continuously.

The x axis represents the change in the weight values in hidden layer. Figure 8 presents the outputs of nodes in the output layer in echo state network when a 3 X 3 window is given as input. The x-axis represents the window number and the y-axis plots node output value. This graph is obtained for hidden layer with 7 reservoirs. The hidden layer with than 7 reservoirs do not produce correct segmented textures.

VII. FUZZY LOGIC

Fuzzy logic is a form of many valued logic; it deals with reasoning that is approximate rather than fixed and exact. In contrast with traditional logic theory, where binary sets have two-valued logic: true or false, fuzzy logic variables may have a truth values that ranges in degree between 0 and 1. Fuzzy logic has been extended to handle the concept of partial truth, where the truth value may range between completely true and completely false. Fuzzy systems are an alternative to traditional notions of set membership and logic.

a. Fuzzy logic is conceptually easy to understand.
b. The mathematical concepts behind fuzzy reasoning are very simple. Fuzzy logic is a more intuitive approach without the far-reaching complexity.
c. Fuzzy logic is flexible.
d. With any given system, it is easy to layer on more functionality without starting again from scratch.
e. Fuzzy logic is tolerant of imprecise data.
f. Everything is imprecise if you look closely enough, but more than that, most things are imprecise even on careful inspection. Fuzzy reasoning builds this

understanding into the process rather than tacking it onto the end.

g. Fuzzy logic can model nonlinear functions of arbitrary complexity.

We can create a fuzzy system to match any set of input-output data. This process is made particularly easy by adaptive techniques like Adaptive Neuro-Fuzzy Inference Systems (ANFIS), which are available in Fuzzy Logic Toolbox software. Fuzzy logic can be built on top of the experience of experts.

In direct contrast to neural networks, which take training data and generate opaque, impenetrable models, fuzzy logic lets you rely on the experience of people who already understand your system. Fuzzy logic can be blended with conventional control techniques. Fuzzy systems don't necessarily replace conventional control methods. In many cases fuzzy systems augment them and simplify their implementation. Fuzzy logic is based on natural language. The basis for fuzzy logic is the basis for human communication. This observation underpins many of the other statements about fuzzy logic. Because fuzzy logic is built on the structures of qualitative description used in everyday language, fuzzy logic is easy to use. The last statement is perhaps the most important one and deserves more discussion. Natural language, which is used by ordinary people on a daily basis, has been shaped by thousands of years of human history to be convenient and efficient. Sentences written in ordinary language represent a triumph of efficient communication.

The training and testing fuzzy logic is to map the input pattern with target output data. For this, the inbuilt function has to prepare membership table and finally a set of number is stored. During testing, the membership function is used to test the pattern.

Training Fuzzy logic

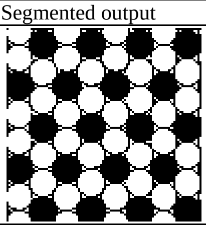
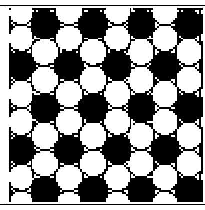
Step 1: Read the pattern (texture image feature) and its target value.
Step 2: Create Fuzzy membership function.
Step 3: Create clustering using K-Means algorithm.
Step 4: Process with target values.
Step 5: Obtain final weights.

Testing Fuzzy logic for texture segmentation

Step 1: Input a pattern (texture image feature).
Step 2: Process with Fuzzy membership function.
Step 5: Find the cluster to which the pattern belongs.
Step 4: Obtain estimated target values.
Step 5: Classify the texture

Table 3 presents the segmented texture outputs for two different radii.

Table 3 Outputs of Fuzzy logic

Radii in Fuzzy logic	Segmented output
0.1(Figure 9)	
0.6(Figure 10)	

RADII specifies the range of influence of the cluster center for each input and output dimension, assuming the data falls within a unit hyperbox (range [0 1]). Specifying a smaller cluster radius will usually yield more, smaller clusters in the data, and hence more rules. When RADII is a scalar it is applied to all input and output dimensions.

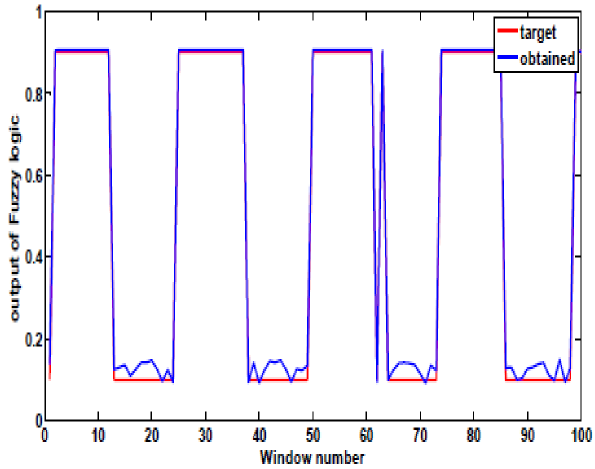


Figure. 9 Output of Fuzzy logic (Radii=0.1)

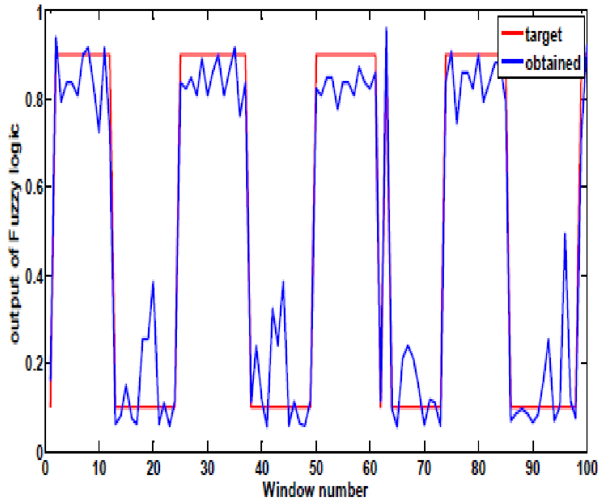


Figure.10 Output of Fuzzy logic (Radii= 0.6)

VIII. CONCLUSION

This paper presents the implementation of echo state neural network for segmentation of textile textures. In the paper, textile textures and mosaic will be presented. The conclusions of this research work are as follows:

a. The recurrent echo state neural network is a promising method for segmenting a given texture. The quality of texture depends upon the number of reservoirs.
b. The output of fuzzy logic depends upon the radii of the clusters used during segmentation.

IX. REFERENCES

[1]. Purushothaman S. and Suganthi D., 2008, Vol. 2, No. 1, pp. 1-9, "FMRI segmentation using echo state neural network", International Journal of Image Processing-CSI Journal

[2]. Shoudong Han, Wenbing Tao, Xianglin Wu, 2011,Texture segmentation using independent-scale component-wise Riemannian-covariance Gaussian mixture model in KL measure based multi-scale nonlinear structure tensor space, Pattern Recognition, v.44 n.3, p.503-518

[3]. Varma, M. and Zisserman, A., A statistical approach to texture classification from single images , International Journal of Computer Vision - Special Issue on Texture Analysis and Synthesis archive Volume 62 Issue 1-2, April-May 2005.

[4]. Varma, M. and Zisserman, A., Texture Classification: Are Filter Banks Necessary?, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2003).

[5]. Varma, M. and Zisserman, A.Classifying Images of Materials: Achieving Viewpoint and Illumination Independence Proceedings of the 7th European Conference on Computer Vision, Copenhagen, Denmark (2002).

[6]. Won W. K. g**,** C. W. M. Yuen D. D. Fan L. K. Chan E. H. K. Fung**,** 2009**,** Stitching defect detection and classification using wavelet transform and BP neural network, Expert Systems with Applications: An International Journal, Volume 36 Issue 2.

[7]. Yining Deng, b. s. Manjunath, 2001, Unsupervised Segmentation of Color-Texture Regions in Images and Video, IEEE Transactions on Pattern Analysis and Machine Intelligence, v.23 n.8, p.800-810

X. BIOGRAPHIES

1. Shoba Rani received the MCA degree from Bharathidasan University. She worked as a Lecturer in LWIASE, Chennai. Also, she completed M.Phil. from Annamalai University. After that she worked for two years as a Consultant in Computer Science Dept. for Tamil Nadu Open University, Chennai. Presently she is working as a Assist. Prof in Dr. MGR University, Chennai. She is also a Ph.D. research scholar in the department of Computer Science at Mother Teresa Women's University, Kodaikanal.

2. Dr. S. Purushothaman, is presently working as Principal, Sun College of Engineering and Technology, Nagercoil. He is obtained his Ph.D. degree from IIT (M), Chennai. He has so far successfully guided nearly six Ph.D. Research Scholars.