ISSN:0975-9646

Rudranath Banerjee et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (6) , 2014, 7002-7011

Image Segmentation using Bi Directional Self Organize Neural Network (BDSONN)

Rudranath Banerjee[1] and Arun Kanti Manna[2]

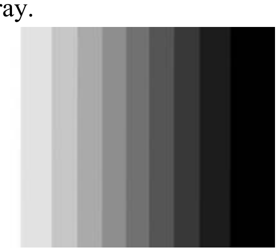
Dept. of CSE,Techno India University ,Kolkata,India.

Abstract-An image segmentation using bi-directional self-organizing neural network (BDSONN) with Multilevel Sigmoidal (MUSIG) function as activation function approach is presented in this Report to segment image in different color levels(for color images) or gray levels(for gray scale images). Multilevel sigmoidal (MUSIG) activation function induces multi scaling capabilities in bi-directional self-organizing neural network (BDSONN) architecture. The function however resorts to equal and fixed class responses, assuming the homogeneity of image information content. The color and gray scale images are segmented by applying the resultant multilevel sigmoidal (MUSIG) activation function. Results of segmentation of color and gray images indicate segmentation efficiency of the BDSONN using MUSIG function.

Keywords - Image segmentation, Bi-Directional self-organizing neural network (BDSONN), MUSIG activation function.

1. INTRODUCTION

Image processing [3] is any form of signal processing for which the input is an image, such as photographs; the output of image processing can be either an Image or a set of characteristics or parameters related to the image. Most image processing techniques involve treating the image as array, or a matrix, or a square pixel arranged in column and row. In 8 bits grayscale image each pixel has assigned intensity from 0 to 255.Gray scale image is what people normally call black and white image, but the name emphasize that such an image also include many shades of gray.



A normal grayscale image has 8 bit color depth=256 grayscales. Where as a true color image has 24 bit color depth=8 x 8 x 8 bits=256 x 256 x256= 16 million colors

Digital image processing is the use of computer algorithms to perform image processing on digital images. It allows a much wider range of algorithms to be applied to the input data, and can avoid problems such as the build-up of noise and signal distortion during processing.

Digital image processing methods were introduced in 1920, when people were interested in transmitting picture information across the Atlantic Ocean.

The various steps required for any digital image processing applications are listed

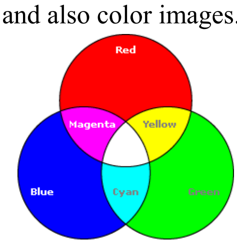
Below:-

I. Image acquisition
II. Preprocessing
III. Segmentation
IV. Representation and feature extraction
V. Recognition and interpretation

An image is digitized to convert it to a form, which can be stored in a computer's memory or on some form of storage media such as a hard disk or CD-ROM. This digitization procedure can be done by a scanner, or by a video camera connected to a frame grabber board in a computer. Once the image has been digitized, it can be operated upon by various image-processing operations.

Imaging set-up (for example, bad lighting) can be corrected using Image Enhancement techniques. Once the image is in good condition, the Measurement Extraction operations can be used to obtain useful information from the image.

Image processing involves processing or altering an existing image in a desired manner. The first step is obtaining an image in a readable format. This is much easier today than five years back. Once the image is in a readable format, image processing software needs to read it so it can be processed and written back to a file. An image consists of a two-dimensional array of numbers. The color or gray shade displayed for a given picture element (pixel) depends on the number stored in the array for that pixel. It is a binary image since each pixel is either 0 or 1. People can distinguish about 40 shades of gray, so a 256 shade image looks like a photograph. This project concentrates on gray scale images and also color images.



The most complex type of image is color. Color images are similar to gray scale except hat there are three bands, or channels, corresponding to the colors red, green, and blue. All though there is so many color model, but the RGB is the popular one Thus, each pixel has three values associated with it.It use additive color mixer and is a basic color model used in television and any other miduam that projects color with light.RGB color model also used in computer and for web graphics but it cannot be used for print production.the secondary color of RGB is cyan, magenta and yellow. A color scanner uses red, green, and blue filters to produce those values.

Images are available via the Internet, scanners, and digital cameras. Any picture shown on the Internet can be downloaded by pressing the right mouse button when the pointer is on the image. This brings the image to the PC usually in a JPEG format. Your software packages can convert that to correct resolution. Image scanners permit putting common photographs into computer files.

2. DIGITAL IMAGE REPRESENTATION

The term monochrome or simply, refers to a two-dimensional intensity function *f(x,y),*where *x* and *y* denote spatial coordinates and the value of *f* at any point *(x,y)* is proportional to the brightness (or the gray level)of the image at that point. Sometimes viewing an image functioning perspective with the third axis being brightness is useful. In this way as series of active peaks in regions with numerous changes in brightness levels and smoother regions or plateaus where the brightness levels varied little or were constant. Using the convention of assigning proportionately higher values to brighter areas would make the height of the components in the plot proportional to the corresponding brightness of the image.

A digital image [11] is an image *f(x, y)* that has been discretized both in spatial coordinates and brightness. A digital image can be considered a matrix whose row and column indices identify a point in the image and the corresponding matrix element value identifies the gray level at the point .the elements of the digital array are called ***image elements, picture elements, pixels or pels,*** with the last two being commonly used abbreviations of "picture elements".

Although the size of a digital image varies with the application, demonstrate the many advantages of selecting square arrays with sizes and number of gray levels that are integer powers of 2. For example, a typical size comparable in quality to a monochrome image is a 512 *512 array with 128 gray levels.

15	0	0	0	0	15
15	3	3	3	3	15
15	3	0	0	3	15
15	3	0	0	3	15
15	3	0	0	3	15
15	3	3	3	3	15
15	0	0	0	0	15

The above zoomed picture is a *6 X 7* picture and its pixel value is written in a matrix form.

Here, *x*=7 and *y*=6.Then we can find *f (2, 2) =3, f (4, 3) =0* and so on.

2.1 Simple image model:

The term image [3] refers to a two-dimensional light-intensity function, denoted by *f(x, y)*, where the value or amplitude of f at spatial coordinates *(x, y)* gives the intensity (brightness) of the image at that point. As light is a form of energy *f(*x, y) must be nonzero and finite that is,

$$0< f(x, y) <\infty \quad ---------- (1)$$

The basic nature of f(x, y) may be characterized by two components:

The amount of source light incident on scene is being viewed. The amount light reflected by the objects in scene. Appropriately, they are called the illumination and reflectance components, and are denoted by *i(x, y)* and *r(x, y),* respectively. The functions *i(x, y)* and *r(x, y)* are combine as a product to form *f(x, y)*:

$$f(x, y) = i(x, y)r(x, y) \quad -----------(2)$$

where

$$0 < i(x, y) < \infty \quad ---------- (3)$$

and

$$0 < r(x, y) < 1 \quad ---------- (4)$$

Equation (4) indicates is bounded by 0 (total absorption) and 1 (total reflectance). The nature of *i(x, y)* is determined by the light source, and *r(x, y)* is determined by the characteristics of the objects in a scene.

The values given in Equations (3) and (4) are theoretic bounds. The following average numerical figures illustrate some typical ranges of *i(x, y)*. On a clear day, the sun may produce in excess of 9000 foot-candles of illumination on the surface of the earth. The figure decreases to less than 1000 foot-candles on cloudy day. On a clear evening, a full moon yields about 0.01 foot-candle of illumination. The typical illumination level in a commercial office is about 100 foot-candles. Similarly the following are some typical values of *r(x, y)*: 0.01 for black velvet, 0.65 for stainless steel, 0.80 for flat-white wall paint, 0.90 for silver plated metal, and 0.93 for snow.

The intensity of a monochrome image f at coordinates *(x, y)* the gray level (l) of the image of the point. From equations (2) through (4), it is evident that l lies in the range

$$L_{min} \leq l \leq L_{max}$$

The only requirement on $L_{min} = i_{min}\, r_{min}$ and $L_{max} = i_{max}\, r_{max}$. Using the preceding values of illumination and reflectance, the values $L_{min}=0.005$ and $L_{max}=100$ for indoor image processing applications may be expected.

The internal $[L_{min}, L_{max}]$ is called the gray scale. This interval numerically to the interval *[0, L]*, where *l=0* is considered black and *l=L* is considered white in the scale. All intermediate values are shades of gray varying continuously from black to white.

3. IMAGE PROCESSING

Digital image processing [11] encompasses a broad range of hardware, software and theoretical underpinnings and it will be helpful to use a 'theme'.

An application that is rather easy to conceptualize without any prior knowledge of imaging concepts is the use of image processing techniques for automatically reading the address on pieces of mail. The overall objective is to produce a result from a problem domain by means of image processing. The problem domain consists of pieces of mail, and the objective is to read the address on each piece. Thus the desired output in this case is a stream of alphanumeric characters.

Step1: The first step in the process is image acquisition- that is, to acquire a digital image. To do so requires an imaging sensor and the capability to digitalize the signal produced by
the sensor. The imaging sensor could also be a line-scan camera that produces a single image line at a time.

Step2: The next step deals with preprocessing that image. The key function of preprocessing is to improve the image in ways that increase the chances for success of the other process. Preprocessing typically deals with technique for enhancing contrast, removing noise, and isolating regions whose texture indicate a likelihood of alphanumeric information.

Step3: The next step deals with segmentation. Segmentation partitions an input image into its constituent parts or objects. In general autonomous segmentation is one of the most difficult tasks in digital image processing. On the one hand, a rugged segmentation procedure brings the process a long way toward successful solution of an imaging problem. On the other hand weak or erratic segmentation algorithms almost always guarantee eventual failure. In the terms of character recognition, the key role of segmentation is to extract individual characters and words from the background. The output of the raw pixel data, constituting either the boundary of a region or all the points in the region itself. In either case, converting the data to a form suitable for computer processing is necessary. The first decision that must be made is whether the data should be represented as a boundary or as complete region. Boundary representation is appropriate when the focus is on external shape characteristics, such as corners and inflections. Regional representation is appropriate is on internal properties, such as texture or skeletal shape. In some applications however, these representations coexist. This situation occurs in character recognition applications, which often require algorithms based on boundary shape as well as skeletons other internal properties.

Step4: Choosing a representation is only part of the solution for transforming raw data into a form suitable for subsequent computer processing. A method must also be specified for describing the data for features of interest are highlighted. *Description*, also called *feature selection*, deals with extracting features that result in some quantitative information of interest or features that are basic for differentiating one class of objects from another.

Step5: The last stage is recognition and interpretation. Recognition is the process that assigns a label to an object based on the information provided by its descriptors. Interpretations involve assigning meaning to an ensemble of recognized objects.

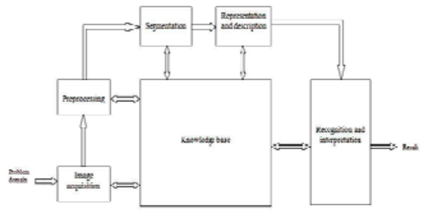


Fig: Fundamental Steps in Digital Image Processing [7]

4. IMAGE SEGMENTATION

Segmentation [11] refers to the process of partitioning a digital image into multiple region (set of pixels). Each of the pixels in a region are similar with respect to some characteristic such as color, intensity, texture. Adjacent regions are different with respect to the same.

4.1 Gradient operator:

This is a part of image segmentation [4]. Through the gradient operator the edge can be detected and make it smooth.

For estimating image gradients from the input image or a smoothed version of it, different gradient operators can be applied. The simplest approach is to use central differences: [4]

$$L_x(x,y) = -1/2 L(x-1,y) + 0 L(x,y) + 1/2 L(x+1,y).$$

$$L_y(x,y) = -1/2 L(x,y-1) + 0 L(x,y) + 1/2 L(x,y+1).$$

4.2 Edge Detection:

There are many methods for edge detection, but most of them can be grouped into two categories, search-based and zero-crossing based. The search-based methods detect edges by first computing a measure of edge strength, usually a first-order derivative expression such as the gradient magnitude, and then searching for local directional maxima of the gradient magnitude using a computed estimate of the local orientation of the edge, usually the gradient direction. The zero-crossing based methods search for zero crossings in a second-order derivative expression computed from the image in order to find edges, usually the zero-crossings of the Laplacian or the zero crossings of a non-linear differential expression, as will be described in the section on differential edge detection following below. As a pre-processing step to edge detection, a smoothing stage, typically Gaussian smoothing, is almost always applied.

The well-known and earlier Sobel operator is based on the following filters: [4]

$$L_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * L \quad \text{and} \quad L_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * L$$

Given such estimates of first- order derivatives, the gradient magnitude is then computed as:

$$|\nabla L| = \sqrt{L_x^{\,2} + L_y^{\,2}}$$

While the gradient orientation can be estimated as

$$\theta = a\tan 2(L_y, L_x)$$

4.3 Point Detection:

The detection of isolated points in an image is straightforward. Using the mask shown in next Fig, we say that a point has been detected at the location on which the mask is centered if $R > T$ ……………… (1)

Where T is a nonnegative threshold and R is given by the equation below [7]

$$R=W_1Z_1 + W_2Z_2+ W_3Z_3+\ldots..+ W_9Z_9$$

Where Z_i is the gray level of the pixel associated with mask coefficient W_i.

Basically, all that this formulation does is measure the weighted differences between the center point and its neighbors. The idea is that the gray level of an isolated point will be quite different from the gray level of its neighbors.

4.4 Region–oriented segmentation:

The objective of segmentation is to partition an image into regions. We approach this problem by finding boundaries between regions based on intensity discontinuities, whereas earlier segmentation was accomplished via thresholds based on the distribution of pixel properties, such as intensity or color. Here we discussed segmentation techniques that are based on finding the regions directly.

Basic formulation:

Let R represent the entire image region. We may view segmentation as a process that partitions R into n sub-regions, $R_1, R_2 \ldots R_n$, such that

a. $\sum_{i=0}^{n} R_i = R$,

b. R_i is a connected region, $i = 1,2,\ldots\ldots.n$,

c. $R_i \cap R_j = \phi$ for all i and j, $i \neq j$,

d. $P(R_i)$ = TRUE for $i = 1,2,\ldots\ldots\ldots.,n$, and

e. $P(R_i \cup R_j)$ = FALSE for $i \neq j$,

Where $P(R_i)$ is a logical predicate over the points in set R_i and ϕ is the null set.

Condition (a) indicates that the segmentation must be complete; that is, every pixel must be in a region. The second condition requires that points in a region must be connected. Condition (c) indicates that the regions must be disjoint. Condition (d) deals with the properties that must be satisfied by the by the pixels in a segmented region- for example $P(R_i)$ =TRUE if all pixels in R_i have the same intensity. Finally, condition (e) indicates that regions R_i and R_j are different in the sense of predicate P.

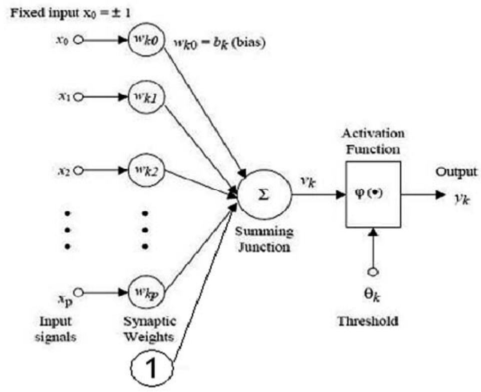
5. NEURAL NETWORK CONCEPT

Neural Network [2] which are simplified models of biological neuron system, is a massively parallel distributed processing system made up of highly interconnected neural computing elements that have the ability to learn and thereby acquire knowledge and make it available for use.

The structural constituents of human brain termed neurons are entity which performs computation such as cognition, logical inference, pattern recognition and so on. Hence the technology, which has been built on a simplified imitation of computing by neuron of a brain, has been termed as Artificial Neural Network.

A human brain develops with time; this common parlance is known as experience. Technically this involves the development of neuron to adapt to their surrounding environment by updating their weight factors.

5.1 BASIC MODEL OF ARTIFICIAL NEURAL NETWORK [8]



Here $x1, x2, x3\ldots.xn$ are the input to artificial neuron.$w1,w2,w3,w4\ldots wn$ are the corresponding weights attached to the input links. One biased input has been considered having weight $-k$.

So the output can be calculated as bellow..

$$Vk=Wk1.X1+Wk2.X2+Wk3.X3-k$$

$$Vk = \sum_{i=1}^{n} X_i.Wk_i.-k$$

Now the activation function wll be applied on it.

$$Vk' = f\sum_{i=1}^{n} X_i W k_i . - k$$

5.2. COMPONENT, STRUCTURE AND TOPOLOGY OF NEURAL NETWORK

Neurons: Nuron are the basic computational elements of the neural network.generally there are three types of neurons in neural network architecture.These are input, hidden, and output neurons. Input neuron accept real world data,Hidden layers lie on the middele of input and output layer.All the computation is being done by hidden neurons.output neuron represent the output. [10]

Neuron Activation state vector: The transmission of information through different neurons of human nervous system depends on the activation level of nerve cell or neurons.If there is n numbers of nerves then the activation state vector will be [10]

$$X = \begin{bmatrix} X1 \\ X2 \\ X3 \\ . \\ . \\ Xn \end{bmatrix}$$

Activation Transfer Function: This function represent the transfer characteristic of individual neuron.It determines the behavior of the incident inputs using dutiable learning algorithem.

Interconnection topology: This refers the mode of interconnection of the neuron in different layers of neural network.As in the biological neuron axon synopsis interconnection act as the storage junction similarly the interconnection if artificial neural network is characterized by weight factors which ascertain their activation level.

Learning Algorithem: There are two types of learning one is feedforward manner and another is feedback manner where loop exist.In feedback algorithem information can be moved in backward direction or to the previous layers.

5.3. OPERATION MODES:

The operation modes of neural network can be classified into two main categories viz, supervised ans unsupervised learning.

Supervised learning:

This architecture implies the use of some prior knowledge base to guide the learning phase. This knowledge base puts forward to the neural network a training set of input-output patterns and an input –output relationship. The neural network is then supervised to embed an approximation function in its operation.

Unsuprvised learning:

This is an adaptive learning paradigm, which present the neural network with an input and allow it to self-organize the topological configuration depending on the distribution of the input data by means of prototype of input vector presented.

Mathematical formulation:

Neural network is nothing but a vector matrix multiplication in simple sense. If there are *n* numbers of inputs like *{q1,q2,q3,q4….,qn}* arriving in at the receiving neuron and the interconnection weights matrix from m such receiving neurons to *n* preceding neuron is represented by *Wmxn={wij,i=1,2,3,...n,j=1,2,3,....m}* then the summing mechanism is carried out by

$$O = \begin{vmatrix} w11 & w12 & w13 & . & . & . & w1n \\ w21 & w22 & w23 & . & . & . & w2n \\ . & . & . & . & . & . & w3n \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ wm1 & wm2 & . & . & . & . & wmn \end{vmatrix} X \begin{vmatrix} q1 \\ q2 \\ q3 \\ . \\ . \\ . \\ qn \end{vmatrix}$$

Then the sutible activation function will be applied on it to gate the expected output.

$$O^{'} = f \begin{vmatrix} w11 & w12 & w13 & . & . & . & w1n \\ w21 & w22 & w23 & . & . & . & w2n \\ . & . & . & . & . & . & w3n \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ wm1 & wm2 & . & . & . & . & wmn \end{vmatrix} X \begin{vmatrix} q1 \\ q2 \\ q3 \\ . \\ . \\ . \\ qn \end{vmatrix}$$

Multilayer Perceptrons:

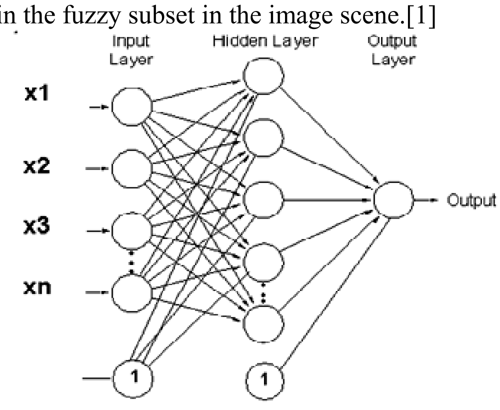
As the single layer perceptrons is not capable to work with the non linear set of data so multi layered perceptrons is introduced. Here an intermediate layer between input and output layer is considered, known as hidden layer. The three layered neural network architecture is given bellow. [8]

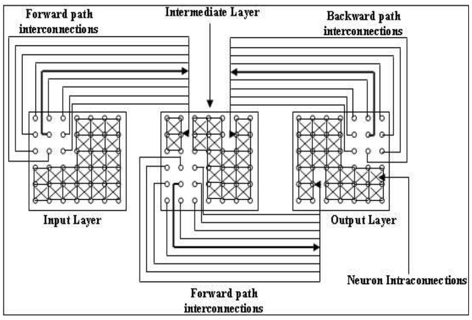
6. IMPLEMENTATION

Image segmentation and classification is a challenging task in the image processing fraternity owing to the varity and complexity associated therein.The problem of image segmentation become more uncertain and severe when it comes to color image segmentation.

Here the neural architecture used comprises of three layers bi directional self organizing neural network(BDSONN) architecture [1] comprising fully connected neurons for the extraction of the object from the noisy image and capable of incorporating the underlying image context heterogeneity through variable and adaptive thresholding.

The input layer of the network represent the fuzzy membership information of the image scene to extracted.the second layer and the output layer or final layer deals with self supervised object extraction task by bi-directional propagation of the network state. Each layer except the output layer is connected to the next layer following the neighborhood based topology. The output layer neurons in turns, connected to the intermediate layer following the similar topology, thus forming counter-propagation architecture with the intermediate layer. The novelty of the proposed architecture is that the assign or updating of the inter layer connection weights are done using the relative fuzzy membership value at the constituent neuron in the different layers. The inter connection strength is decided by the relative fuzzy membership values of the neighbors and the candidate neuron and there by influenced by the local heterogeneity within the fuzzy subset in the image scene.[1]





If µkj is the membership value at the jth candidate neuron in the kth layer and µkj is the membership value at its ith second order neighbor in the same layer, then the *inter*-layer connection strength, Wkilj, between the corresponding candidate neuron of the next lth layer and the ith second order neighbors of the kth layer is given by [1]

$$w_{kilj} = \overline{\mu_{kj} - \mu_{ki}}$$

The output layer neurons are similarly connected to the intermediate layer neurons in the backward direction. In addition, there is also fixed and full connectivity between the corresponding neurons of the different layers of the network. If I_{ki} are the fuzzy membership values at ith neighbors of the k^{th} layer neurons, then the input at the j^{th} neuron of the next l^{th} layer, which enjoys connectivity with this k^{th} layer neighbourhood, is given by

$$I_{ij} = \sum_i w_{kilj} I_{ki}$$

Where, W_{kilj} are the *inter*-layer interconnection weights. The output,Oj , produced by this neuron is given by

$$O_j = f(I_{ij})$$

Where,f is the beta activation function with context sensitive thresholding and is given as [1]

$$f(t) = \int_0^t K x^{\alpha} (1-x)^{\beta c}\, dx, \alpha, \beta c \geq 0, t \in [0,1]$$

Where, t and K have their usual signifi cances. The parameter is the fuzzy cardinality estimate (ξ) of the image neighborhood fuzzy subsets. The resultant context sensitive threshold parameter, (τ_c), which takes into account the image neighborhood intensity distribution through the fuzzy cardinality βc estimates of the neighborhood fuzzy subsets in the form of the parameter is given by

$$\tau_c = \frac{\alpha}{\alpha + \beta c}$$

The choice of the thresholding parameter for the activation function helps in incorporating the image heterogeneity information in the operational characteristics of the network architecture, which otherwise, would be lacking if a single point fi xed thresholding parameter is chosen. As a result, noise immunity and generalization capability are induced in the network architecture. The different values of the threshold parameter corresponding to the different neighborhood fuzzy subsets in the image information are propagated to the succeeding layers of the network using the fixed and full *inter-layer* interconnections between the corresponding neurons of the different layers of the network. In this way, the network input states are propagated from the input layer to the output layer of the network. The backward path *inter-layer* connection strengths from the output layer to the intermediate layer are again evaluated from the relative measures of the fuzzy membership values at the output layer neurons. The output layer network states and the corresponding output layer neighborhood context information are propagated to the intermediate layer through the backward path *inter-layer* connections for further processing. This *to* and *from* propagation of the network states between the two inner layers of the network architecture is continued until the *inter-layer* connection strengths from the intermediate layer to the output layer and back stabilize. At this point, the fuzzy hostility indices, which are reflective of the heterogeneity of the image information content, are reduced to minimum and the original input image information is self supervised into homogeneous object and background regions at the network output layer.

6.1 MULTILEVEL SIGMOIDAL (MUSIG) ACTIVATION FUNCTION:

The multilevel sigmoidal (MUSIG) [9] activation function is capable of generat multileveloutput corresponding to the scales of gray.It is given by [9]

$$f_{MUSIG}(x; \xi_\beta, cl_\beta) = \sum_{\beta=1}^{k-1} \frac{1}{\xi_\beta + e^{-\lambda[x-(\beta-1)cl_\beta - \theta]}}$$

Where ξ_β represent the multilevel class is given by [9]

$$\xi_\beta = \frac{C_N}{cl_\beta - cl_{\beta-1}}$$

Here β represent the gray scale object index and k is the number of gray scale objects.The gray scale contribution of $(\beta-1)^{th}$ and β^{th} classes is denoted by cl_β and the maximum fuzzy membership gray intensity contribution of pixel is denoted by C_n and the threshold parameter is θ in the MUSIG activation function is fixed and uniform.

6.2 ALGORITHM:

Network self-organization algorithm [1] :

The self-supervised operation of the proposed bi-directional self-organizing neural network (BDSONN) architecture [1] comprises four phases viz. (i) network initialization phase, where the *intra*-layer interconnections within the different network layers are initialized to 1, (ii) an input phase, where external world input noisy image scenes are fed at the input layer of the network, (ii) forward propagation phase, where the processed outputs of the network input layer are propagated to the following network intermediate layer and the processed outputs of the network intermediate layer are propagated to the following network output layer, and (iii) backward propagation phase, where network output layer outputs are propagated to the network intermediate layer. Each of the propagation phases are preceded by the determination of the fuzzy cardinality estimates of the neighborhood fuzzy subsets for computing the fuzzy context sensitive thresholding information required for the processing operation of the succeeding network layer. The entire network operation can be summarized by the following algorithm.

1 Begin

Initialization phase

2 Initialize intra_conn[l], l=1, 2, 3

Remark: intra conn[l] are the *intra*-layer interconnection matrices for the three *l* network layers. All *intra*-layer interconnections are set to unity.

Input phase

3 Read pix[l][m][n]

Remark: p[l][m][n] are the fuzzifi ed image pixel information at row *m* and column *n* at the ith network layer, i.e. the fuzzy membership values of the pixel intensities in the image scene. p[1][m][n] are the fuzzy membership information of the input image scene and are fed as inputs to the input layer of the network. p[2][m][n] and p[3][m][n] are the corresponding information at the intermediate and output layers.

Forward propagation phase

4 tauC[l+1][m][n]=f(card[l][m][n])

5 p[l+1][m][n]=fbeta(p[l][m][n] x
wt[t][l][l+1])

Remark: tauC[l+1][m][n] are the adaptive fuzzy context sensitive thresholding information for the (l+1)th network layer neurons. It is a function of card[l][m][n], the corresponding fuzzy cardinality estimates. fbeta is the standard beta activation function and wt[t][l][l+1] are the *inter*-layer interconnection weights between the lth and (l+1)th network layers at aparticular epoch *(t)*, determined from the relative pix[l][m][n] values. The fuzzy context sensitive threshold values and the processed image information are propagated to the following layer (until the output layer is reached) using the *inter*-layer interconnections.

Do

6 Repeat steps 4 and 5 with intermediate
layer outputs

Backward propagation phase

7 tauC[l-1][m][n]=f(card[l][m][n])

8 p[l-1][m][n]=fbeta(p[l][m][n] x
wt[l][l-1])

Remark: Propagation of the adaptive context sensitive threshold values and the processed information in the reverse direction from the network output layer to the network intermediate layer.

Loop

Until((wt[t][l][l-1]-wt[t-1][l][l-1])<eps)

Remark: eps is the tolerable error.

End

6.3 Correlation coefficient (ρ)

The standard measure of correlation coefficient (ρ) [9] can be used to assess the quality of segmentation achieved. It is given by [9]

$$\rho = \frac{\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(A_{ij}-\overline{A})(B_{ij}-\overline{B})}{\sqrt{\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(A_{ij}-\overline{A})}\sqrt{\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(B_{ij}-\overline{B})}}$$

where A_{ij}; $1 \le i; j \le n$ and B_{ij}; $1 < i; j < n$ are the original and the segmented images respectively, each of dimensions n x n. $\bar{A}$ and B are their respective mean intensity values. A higher value of implies better quality of segmentation. However, correlation coefficient has many limitations. The foremost disadvantage is that it is computationally intensive. This often confines its usefulness for image registration, i.e. orienting and positioning two images so that they overlap. Moreover, the correlation coefficient is very much sensible to image skewing, fading, etc. that inevitably occur in imaging systems.

6.4 Result:

Gray Scale Image

Lena Image:-

Input Image:-

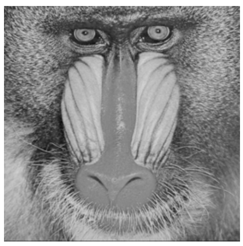
No.	Grey Scale Boundaries	Correlation
P-a	{0, 21, 65, 106, 138, 167, 227, 255}	0.8970
P-b	{0, 41, 80, 123, 158, 201, 240, 255}	0.8846
P-c	{0, 38, 71, 113, 135, 168, 209, 255}	0.9068
P-d	{0, 47, 80, 111, 146, 168, 212, 255}	0.9053
Q-a	{0, 47, 61, 111, 120, 169, 172, 255}	0.8605
Q-b	{0, 47, 53, 117, 119, 167, 172, 255}	0.7994
Q-c	{0, 48, 55, 121, 135, 185, 195, 255}	0.8051
Q-c	{0, 53, 55, 119, 121, 170, 186, 255}	0.7878

Table 1:Output result of gray Lena Image

The output images are shown in the figure 6.4.1 and 6.4.2. figure 6.4.1 (a) represents the output image corresponding the set of serial no P-a. Figure 6.4.1(b) represents the output image corresponding the set of serial no P-b and so on. Similerly figure 6.4.2 (a) represents the output image corresponding the set of serial no Q-a and figure 6.4.2 (b) represents the output image corresponding the set of serial no Q-b and so on.

Baboon Image :-

Input Image :-



No.	Grey Scale Boundaries	Correlation
P-a	{0, 68, 91, 112, 141, 169, 190, 255}	0.7570
P-b	{0, 66, 84, 116, 143, 166, 189, 255}	0.7537
P-c	{0, 65, 104, 128, 152, 168, 191, 255}	0.7720
P-d	{0, 40, 78, 114, 140, 164, 191, 255}	0.7543
Q-a	{0, 82, 90, 125, 133, 163, 169, 255}	0.6919
Q-b	{0, 81, 82, 124, 126, 159, 162, 255}	0.6464
Q-c	{0, 76, 82, 108, 114, 147, 150, 255}	0.6754
Q-c	{0, 85, 86, 125, 132, 164, 173, 255}	0.6673

Table 2:Output result of gray Baboon Image

The output images are shown in the figure 6.4.3 and 6.4.4. figure 6.4.3 (a) represents the output image corresponding the set of serial no P-a. Figure 6.4.3(b) represents the output image corresponding the set of serial no P-b and so on. Similerly figure 6.4.4 (a) represents the output image corresponding the set of serial no Q-a and figure 6.4.5 (b) represents the output image corresponding the set of serial no Qbb and so on.

RGB Image:-

Lena Image:-

Input Image:

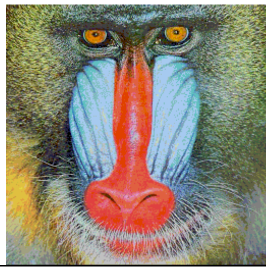
No	Bounderies	Correleation
P-a	R={43, 89, 156, 160, 172, 213, 237, 255}	0.9052
	G={0, 34, 67, 110, 151, 155, 187, 255}	
	B ={32, 67, 91, 107, 126, 164, 186, 238}	
P-b	R={43, 70, 120, 130, 150, 170, 190, 255	0.9035
	G={0, 20, 140, 180, 190, 210, 230, 255}	
	B ={32, 45, 77, 85, 115, 145, 175, 238}	
P-c	R={43, 104, 129, 134, 155, 165, 183, 255}	0.9041
	G={0, 13, 57, 147, 160, 207, 221, 255}	
	B={32, 66, 89, 118, 143, 167, 190, 238}	
P-d	R={43, 121, 129, 144, 196, 214, 233, 255}	0.9082
	G={0, 30, 51, 120, 134, 138, 204, 255}	
	B ={32, 65, 77, 101, 123, 154, 187, 238}	
Q-a	R={43, 80, 90, 110, 120, 190, 200, 255}	0.9111
	G={0, 30, 75, 80, 100, 160, 220, 255}	
	B ={32, 50, 70, 90, 110, 150, 170, 238}	
Q-b	R={43, 96, 109, 128, 141, 153, 206, 255}	0.9055
	G={0, 51, 61, 116, 141, 176, 186, 255}	
	B={32, 70, 81, 94, 120, 153, 181, 238}	
Q-c	R={43, 70, 108, 116, 126, 151, 165, 255}	0.9046
	G={0, 41, 111, 157, 166, 180, 200, 255}	
	B={32, 53, 74, 88, 128, 145, 162, 238}	
Q-d	R={43, 58, 72, 140, 172, 210, 232, 255}	0.8947
	G={0, 15, 48, 110, 142, 178, 220, 255}	
	B={32, 55, 87, 112, 136, 140, 215, 238}	

Table 3:Output result of color Lena Image

The output images are shown in the figure 6.4.5 and 6.4.6. figure 6.4.5 (a) represents the output image corresponding the set of serial no P-a. Figure 6.4.5(b) represents the output image corresponding the set of serial no P-b and so on. Similerly figure 6.4.6 (a) represents the output image corresponding the set of serial no Q-a and figure 6.4.6 (b) represents the output image corresponding the set of serial no Q-b and so on.

Baboon Image:-

Input Image:-



No	Bounderies	Correlation
P-a	R={0, 17, 52, 73, 83, 116, 204, 255}	0.8579
	G={0, 27, 102, 152, 153, 180, 254, 255}	
	B ={0, 48, 89, 117, 136, 177, 230, 255}	
P-b	R={0, 82, 85, 159, 163, 210, 214, 255}	0.8541
	G={0, 84, 96, 114, 163, 194, 207, 255}	
	B={0, 30, 69, 100, 124, 153, 211, 255}	
P-c	R={0, 94, 106, 146, 173, 186, 203, 255}	0.8481
	G={0, 39, 69, 74, 88, 111, 147, 255}	
	B={0, 52, 75, 90, 127, 162, 219, 255}	
P-d	R={0, 50, 60, 90, 100, 110, 150, 255}	0.8044
	G={0, 60, 110, 160, 170, 185, 200, 255}	
	B ={0, 10, 20, 60, 80, 150, 220, 255}	
Q-a	R={0, 57, 59, 79, 100, 244, 251, 255}	0.7875
	G={0, 15, 41, 108, 169, 178, 196, 255}	
	B={0, 47, 51, 111, 115, 221, 233, 255}	
Q-b	R={0, 91, 106, 125, 142, 143, 170, 255}	0.8631
	G={0, 52, 54, 75, 83, 86, 167, 255}	
	B={0, 52, 70,90, 126, 160, 223, 255}	
Q-c	R={0, 45, 55, 75, 95, 225, 245, 255}	0.7897
	G={0, 10, 35, 95, 155, 175, 195, 255}	
	B={0, 35, 45, 105, 110, 215, 230, 255}	
Q-d	R={0, 78, 110, 134, 186, 210, 233, 255}	0.8715
	G={0, 42, 68, 92, 120, 162, 240, 255}	
	B={0, 65, 114, 148, 167, 195, 230, 255}	

Table 4:Output result of color Baboon Image

The output images are shown in the figure 6.4.7 and 6.4.8. figure 6.4.7 (a) represents the output image corresponding the set of serial no P-a. Figure 6.4.7(b) represents the output image corresponding the set of serial no P-b and so on. Similerly figure 6.4.8 (a) represents the output image corresponding the set of serial no Q-a and figure 6.4.8 (b) represents the output image corresponding the set of serial no Q-b and so on.

Figure 6.4.1: class segmented test images (gray Lena) with optimized class boundaries referring to Table 1.
(a) Represents image for the boundery P-a, (b) represents image for the boundery P-b,
(c) Represents image for the boundery P-c, (d) represents image for the boundery P-d.

Figure 6.4.2: class segmented test images (gray Lena) with optimized class boundaries referring to Table 1.
(a) Represents image for the boundery Q-a, (b) represents image for the boundery Q-b,
(c) Represents image for the boundery Q-c, (d) represents image for the boundery Q-d.

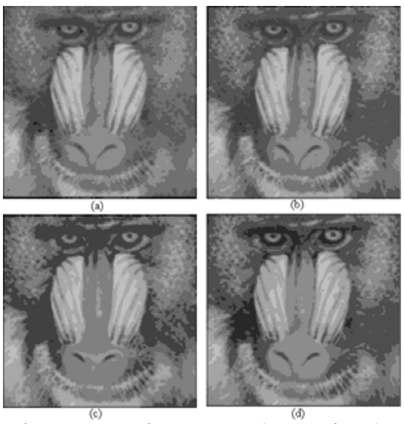


Figure 6.4.3: class segmented test images (gray Baboon) with optimized class boundaries referring Table 2.
(a) Represents the image for boundery P-a, (b) represents the image for boundery P-b,
(c) Represents the image for boundery P-c, (d) represents the image for boundery P-d.

Figure 6.4.4: class segmented test images (gray Baboon) with optimized class boundaries referring Table 2.
(a) Represents image for the boundery Q-a, (b) represents image for the boundery Q-b,
(c) Represents image for the boundery Q-c, (d) represents image for the boundery Q-d.

Figure 6.4.5: class segmented test images (Color Lena) with optimized class boundaries referring Table 3.
(a) Represents the image for boundery P-a, (b) represents the image for boundery P-b,
(c) Represents the image for boundery P-c, (d) represents the image for boundery P-d.

Figure 6.4.6: class segmented test images (Color Lena) with optimized class boundaries referring Table 3.
(a) Represents image for the boundery Q-a, (b) represents image for the boundery Q-b,
(c) Represents image for the boundery Q-c, (d) represents image for the boundery Q-d.

Figure 6.4.7: class segmented test images (Color Baboon) with optimized class boundaries referring Table 4.
(a) Represents the image for boundery P-a, (b) represents the image for boundery P-b,
(c) Represents the image for boundery P-c, (d) represents the image for boundery P-d.

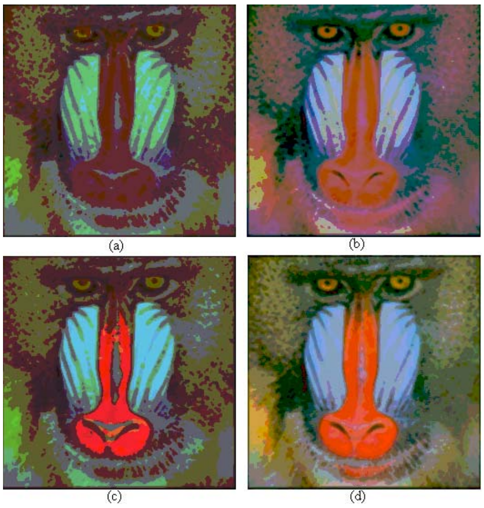


Figure 6.4.8: class segmented test images (Color Baboon) with optimized class boundaries referring Table 4.
(a) Represents image for the boundery Q-a, (b) represents image for the boundery Q-b,
(c) Represents image for the boundery Q-c, (d) represents image for the boundery Q-d.

7. CONCLUSION

The neural network has not been used for the first time to segment an image. Previously image segmentation has been done using the Multi Layer Self-Organizing Neural Network (MLSONN) architecture, which is efficient in binary object extraction from a noisy image through the process of self organization of inputs. The MLSONN architecture has some draw backs. As in this architecture in the back propagation phase there is a recurrent loop connecting the output layer to the input layer which basically increase the complexity. To overcome this draw back Bi-Directional Self-Organizing Neural Network (BDSONN) architecture was introduced [1], in which the output is feed backed to the intermediate layer for minimizing the error. It reduces so many computation burdens as much possible. Image segmentation has been dome using this architecture previously using beta function. But in this project Multi Level Sigmoidal (MUSIG) function has been used as activation function. The main goal of this project was to have a study that MUSIG activation function is efficient or not in case of image segmentation using BDSONN architecture.

However the comparatively study has not been done with the beta function and the MUSIG function.Though the output result shows that MUSIG function is efficient for segmenting an image using BDSONN architecture. So it can be concluded that in coming future MUSIG function will be used widely and will be improved enough for having a better image segmentation using BDSONN architecture.

ACKNOWLEDGMENTS

This work was done by a group of four students of UIT, BU in 2011. The members other than me were- Abhijit Nandi, Subhankar Kabiraj, Soumyajit Ghosh. Prof. Sourav De was our Mentor. I would dedicate this paper to all of my frinds and our project mentor Prof. De.

REFERENCES

[1] S Bhattacharyya, P Dutta, U Maulik and P K Nandi: " A Self Supervised Bi-directional Neural Network(BDSONN) Architecture for Object Extraction Guided by Beta Activation Function and Adaptive Fuzzy Context Sensitive Thresholding" http://www.waset.org/journals/ijece/v1/v1-8-77.pdf

[2] Fundamentals of neural networks a classroom approach-by Satish Kumar

[3] Linda G. Shapiro and George C. Stockman (2001): "Computer Vision", , New Jersey, Prentice-Hall, ISBN 0-13-030796-3

[4] Dzung L. Pham, Chenyang Xu, and Jerry L. Prince (2000): "Current Methods in Medical Image Segmentation", *Annual Review of Biomedical Engineering*, volume 2

[5] Chen C.H., Pau L.F., Wang P.S.P. The Handbook of Pattern Recognition and Computer Vision (2nd Edition). – World Scientific Publishing Co., 1998. – 1004 p.

[6] Vovk, O.L. (2006), "A new approach to visual similar image colors extraction". Journal of Automation and Information Sciences 6: 100-105.

[7] Ron Ohlander, Keith Price, and D. Raj Reddy (1978): "Picture Segmentation Using a Recursive Region Splitting Method", *Computer Graphics and Image Processing*, volume 8

[8] S.Rajasekaram and G.A Vijayalakshmi Pai: "Neural Network Fuzzy logic and Genetics algorithem ,synthesis ans applications"

[9] S. Bhattacharyya, P. Dutta, and U. Maulik, Self organizing neural network (SONN) based grayscale object extractor with a multilevel sigmoidal (MUSIG) activation function, Foundations Comput. Dec. Sci. 33(2) (2008), pp. 131–165.

[10] S Bhattacharyya, "Neural Networks: evaluation, topologies, learning algorithms and applications".

[11] R.C. Gonzalez, R.E. Woods,"Digital Image Processing", Pearson, Prentice Hall.