World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:4, 2008

A Self Supervised Bi-directional Neural Network (BDSONN) Architecture for Object Extraction Guided by Beta Activation Function and Adaptive Fuzzy Context Sensitive Thresholding

Siddhartha Bhattacharyya, Paramartha Dutta, Ujjwal Maulik and Prashanta Kumar Nandi

***Abstract*— A multilayer self organizing neural neural network (MLSONN) architecture for binary object extraction, guided by a beta activation function and characterized by backpropagation of errors estimated from the linear indices of fuzziness of the network output states, is discussed. Since the MLSONN architecture is designed to operate in a single point fixed/uniform thresholding scenario, it does not take into cognizance the heterogeneity of image information in the extraction process. The performance of the MLSONN architecture with representative values of the threshold parameters of the beta activation function employed is also studied. A three layer bi-directional self organizing neural network (BDSONN) architecture comprising fully connected neurons, for the extraction of objects from a noisy background and capable of incorporating the underlying image context heterogeneity through variable and adaptive thresholding, is proposed in this article. The input layer of the network architecture represents the fuzzy membership information of the image scene to be extracted. The second layer (the intermediate layer) and the final layer (the output layer) of the network architecture deal with the self supervised object extraction task by bi-directional propagation of the network states. Each layer except the output layer is connected to the next layer following a neighborhood based topology. The output layer neurons are in turn, connected to the intermediate layer following similar topology, thus forming a counter-propagating architecture with the intermediate layer. The novelty of the proposed architecture is that the assignment/updating of the *inter*-layer connection weights are done using the relative fuzzy membership values at the constituent neurons in the different network layers. Another interesting feature of the network lies in the fact that the processing capabilities of the intermediate and the output layer neurons are guided by a beta activation function, which uses image context sensitive adaptive thresholding arising out of the fuzzy cardinality estimates of the different network neighborhood fuzzy subsets, rather than resorting to fixed and single point thresholding. An application of the proposed architecture for object extraction is demonstrated using a synthetic and a real life image. The extraction efficiency of the proposed network architecture is evaluated by a proposed system transfer index characteristic of the network.

***Keywords*— Beta activation function, fuzzy cardinality, multilayer self organizing neural network, object extraction,

Manuscript received October 28, 2006.

Siddhartha Bhattacharyya is with the Department of Computer Science and Information Technology, University Institute of Technology, The University of Burdwan, Burdwan-713 104, India, phone: +91-342-2558776; fax: +91-342-2558776; email: siddhartha.bhattacharyya@gmail.com.

Paramartha Dutta is with the Department of Computer Science and Engineering, Kalyani Government Engineering College, Kalyani, Nadia-741 235, India, email: paramartha.dutta@gmail.com.

Ujjwal Maulik is with the Department of Computer Science and Engineering, Jadavpur University, Kolkata, India, email: drumaulik@cse.jdvu.ac.in.

Prashanta Kumar Nandi is with the Department of Computer Science and Technology, Bengal Engineering & Science University, Shibpore, Howrah, India, email: nandipkn@rediffmail.com.

I. Introduction

Several image processing applications rely on the extraction and localization of useful object features from the feature space. Removal of noises from image scenes and detection of object regions are no exceptions. A number of attempts in this direction based on classical filtering techniques figure in the literature [1][2][3][4]. Neural networks, assisted by fuzzy logic and optimization tools, often stand useful in these types of applications owing to the inherent capability of handling nonlinear situations [5]. A wide variety of neural networks differing in network architecture have been used by researchers to deal with these types of image preprocessing tasks [6][7][8][9][10][11][12][13]. However, most of these network models rely on external supervision with a set of labeled/classified input data. This mode of supervised operation, which is characteristic of these network models, poses a serious hindrance to the application of these networks in real time situations.

Continuous lookout for network architectures, which can be put to use in real time, has led researchers to evolve newer network architectures, which operate in an unsupervised fashion. These networks self-organize the input information and extract relevant decisions out of the input data. Kohonen's self-organizing feature map [14][15], Hopfield's network [16], the bi-directional associative memory (BAM) [17][18], the cellular neural network [19][20][21] etc. are typical examples of this type of networks. Carpenter *et al.* [22] applied self organizing neural networks for recognition of patterns from images. The multilayer self-organizing neural network architecture (MLSONN) [23], which is a two-dimensional extension of the multilayer perceptron (MLP) [24][25], uses indices of fuzziness of the image information and self-organizes the input information into outputs.

The adjustment of the network interconnection weights in most of these network architectures, is done by the standard backpropagation algorithm at the expense of a greater computational burden. This problem can be alleviated by resorting to such network structures, which do not use the backpropagation based weight adjustment procedure.

Moreover, these networks assume homogeneity in the image information content and hence do not take into account the heterogeneity in the input image information during the application of the characteristic activation. This limitation is supplemented by the fact that the neurons or the processing units of most of these neural network architectures are activated by the standard bilevel sigmoidal activation function [6][23]. The standard sigmoidal activation function is asymptotic in nature and resorts to single uniform thresholding parameters. A bilevel beta activation function, besides being bounded in the interval [0, 1], is also continuously differentiable. The function exhibits sharp and distinct responses to its input variables. In addition, tuning of the function parameters results in generating different thresholding values which is necessary in reflecting the heterogeneity of the image information content. In this article, a fully connected bidirectional three layer self organizing neural network architecture, devoid of backpropagation based weight adjustment and guided by an adaptive thresholded beta activation function, for object extraction from a noisy background, is presented. The architecture comprises an input layer, an intermediate layer and an output layer of neurons. Each layer is interconnected to the next layer following a neighborhood based topology. The output layer is connected to the intermediate layer so as to counter-propagate the network states for the purpose of self supervision. The neurons in each of the three layers are connected to each other within the same layer. Interconnections between the corresponding neurons are present between different layers as well, for the propagation of the fuzzy cardinality estimates, which are indicative of the neighborhood context sensitive information to the other layer neurons. These estimates are used to determine the threshold values of the characteristic beta activation function used for processing. The assignment/updating of the *inter*-layer connection weights are decided by the relative fuzzy memberships of the constituent inputs to the neurons. The network self-organizes the input information by means of counter-propagation of the network states between the intermediate and output layers. The convergence of the network operation is ensured by the stability achieved in the *inter*-layer interconnection weights, which are updated at each stage of processing. The network has been applied for the extraction of objects from several noisy versions of a synthetic image and a real life spanner image. The quality of the extracted images are determined from a systematic point of view by a proposed system transfer index evaluated from the variation in the fuzzy hostility indices in the original, noisy and extracted images, which provides a quantitative measure of the noise immunity factor of the proposed architecture and hence serves as a figure of merit for the network in respect of noise removal.

The paper is organized as follows. Section II presents relevant fuzzy set theoretic concepts and definitions. Fuzzy hostility index for estimation of the network object extraction efficiency, is also introduced in this section. The architecture and operation of the multilayer self organizing neural network (MLSONN) architecture is presented in Section III. The network limitation as regards to incorporation of image content heterogeneity is also demonstrated in this section. The proposed self supervised bi-directional self organizing neural network (BDSONN) architecture characterized by an adaptive image context sensitive thresholding guided beta activation function, is presented in Section IV. The results of application of the bidirectional self organizing neural network architecture on noisy synthetic and real life images are listed in Section V. Section VI concludes the paper with future directions of research.

II. Mathematical Preliminaries

A brief overview of fuzzy set theoretic concepts relevant to the present article and a proposed fuzzy hostility index is presented in this section.

A. Fuzzy set concepts

A fuzzy set A [42][43] contains elements, characterized by a membership function, $\mu_A(x_i)$, where x_i refers to the i^{th} element in the set. This membership function associates with every element in the fuzzy set a membership value, which is indicative of the amount of ambiguity in the fuzzy set. The membership value of an element in a fuzzy set lies in [0, 1]. A higher membership value indicates strict containment of the element within the set, while a lower value indicates weak containment. The support of a fuzzy set S_A of n elements, is defined as the collection of elements having nonzero membership values. Mathematically,

$$S_A = \{\frac{\mu_A(x_i)}{x_i} : \mu_A(x_i) > 0 \text{ and } i = 1, 2, 3, ..., n\} \quad (1)$$

B. Fuzzy set theoretic operations

The following set theoretic operations of union, intersection and complement are defined on two fuzzy sets A, B for an element x in the universe of discourse X.

$$\text{Union} : \mu_{A \cup B} = \max(\mu_A(x), \mu_B(x)) \quad (2)$$

$$\text{Intersection} : \mu_{A \cap B} = \min(\mu_A(x), \mu_B(x)) \quad (3)$$

$$\text{Complement} : \mu_{\overline{A(x)}} = 1 - \mu_A(x) \quad (4)$$

C. Fuzzy cardinality

The scalar cardinality of a fuzzy set A is the sum of the membership values of all the elements in the set. Mathematically, for a fuzzy set A, with a support of n elements, the scalar cardinality, ξ_A is defined as [43]

$$\xi_A = \sum_{i=1}^{n} \mu_A(x) \quad (5)$$

For a finite number of elements, this cardinality is referred to as the fuzzy cardinality. It is evident from this definition that higher is the degree of containment of the elements in the fuzzy set, the higher is the fuzzy cardinality. Similarly, a lower fuzzy cardinality results when the elements are weakly contained in the fuzzy set. Thus, the fuzzy cardinality value of a fuzzy set indicates the overall degree of containment of the constituent elements in the fuzzy set and provides an estimate of the average amount of ambiguity in a fuzzy set. This tantamount to reflecting the variation in the memberships of the constituent elements in the set.

D. Fuzzy entropy measure

The entropy E_A of a fuzzy set A, characterized by membership function $\mu_A(x_i)$ is a measure of the degree of fuzziness in the fuzzy set. For a fuzzy set comprising n elements, it is given by [44]

$$E_A = \frac{1}{nln2}\sum_{i=1}^{n} -plnp - (1-p)ln(1-p) \qquad (6)$$

where $p = \mu_A(x_i)$. The fuzzy entropy measure reflects the amount of ambiguity and corresponds to the randomness/ disorder in an observation.

E. Fuzzy hostility index

In an image, the pixels are surrounded by a number of neighboring pixels in a neighborhood based topology. Depending on the number of neighbors several orders of neighborhood are possible. An image can be viewed as a fuzzy set of pixel intensities, comprising several neighborhood fuzzy subsets of brightness/darkness. The membership values of the elements of such a fuzzy set are proportional to the pixel intensities. Each candidate pixel in a particular neighborhood fuzzy subset is under the influence of its neighbors. The distribution of gray levels of the pixels/intensity membership values in the neighborhood fuzzy subsets, reflects the degree of homogeneity/heterogeneity in that neighborhood subset. The closer are the membership values of a pixel and its neighbors, the higher is the homogeneity in the neighborhood and lesser is the pixel hostile to its neighbors. Sharp deviations in the membership values of the neighborhood pixels lead to a heterogeneous neighborhood in which the candidate pixel is more hostile to its neighbors. This neighborhood homogeneity/heterogeneity in a second order neighborhood can be accounted for by a fuzzy hostility index defined as,

$$\zeta = \frac{3}{8}\sum_{i=1}^{8} \frac{|\mu_p - \mu_{q_i}|}{|\mu_p + 1| + |\mu_{q_i} + 1|} \qquad (7)$$

where, μ_p is the membership of the candidate pixel and μ_{q_i} are those of its neighbors in a second order neighborhood. ζ lies in [0, 1]. A higher value of ζ implies lower neighborhood homogeneity and a lower value of ζ implies higher neighborhood homogeneity. The extrema are obtained with the following representative neighborhood pixel fuzzy memberships.

- If μ_p=0 and $\mu_{q_i} = 0 \; \forall \; i$, then ζ=0 implies the highest neighborhood homogeneity.
- If μ_p=1 and $\mu_{q_i} = 1 \; \forall \; i$, then ζ=0 implies the highest neighborhood homogeneity.
- If μ_p=0 and $\mu_{q_i} = 1 \forall i$, then ζ=1 implies the lowest neighborhood homogeneity.
- If μ_p=1 and $\mu_{q_i} = 0 \forall i$, then ζ=1 implies the lowest neighborhood homogeneity.

III. Multilayer self organizing neural network

The multilayer self organizing neural network (MLSONN) architecture [23] is a feedforward type of network comprising an input layer, any number of hidden layers and an output layer. The number of neurons at the different layers of the network correspond to the number of pixels in the input image scene to be processed. The neurons in a particular layer of the network is connected to the neighbors of the corresponding neurons in the previous layer of the network following a neighborhood based topology. The output layer neurons are in turn connected to the input layer neurons on an one-to-one basis for feedback of outputs. A schematic of the MLSONN architecture is shown in Fig. 1.

A. Operation of the MLSONN architecture

The MLSONN [23] architecture is efficient in extracting binary objects from a noisy image scene through the process of self organization of inputs. The input layer of the network architecture accepts the fuzzy membership information of the input information to be processed. These inputs are propagated to the succeeding layers of the network for further processing. If I_{li} are the inputs to the i^{th} neuron in the l^{th} layer of the network, then the inputs, $I_{(l+1)_k}$ to the k^{th} neuron in the $(l+1)^{th}$ layer of the network is given by

$$I_{(l+1)k} = f(\sum_m I_{li} w_{imk}) \qquad (8)$$

where w_{imk} are the interconnection weights between the k^{th} neuron in the $(l+1)^{th}$ layer and the m neighbors of the i^{th} neuron in the l^{th} layer of the network. f is the standard sigmoidal activation function given by

$$y = f(x) = \frac{1}{1 + e^{(-x-\theta)/\theta_0}} \qquad (9)$$

where θ is a single point uniform threshold parameter and θ_0 controls the steepness of the function.
In this way, the inputs are processed and propagated from one layer to the next until the output layer is reached. Since the MLSONN architecture operates in a self supervised fashion, the network system errors are determined by means of the linear indices of fuzziness in the output layer outputs considering them as a fuzzy set of brightness. The derived system errors are then used to adjust the interconnection weights between the different layers of the network architecture by means of the standard backpropagation algorithm. The outputs are then fed back to the input layer for the next stage of processing with the new set of adjusted weights. This process of self supervision is continued until the network system errors are reduced to some tolerable limit. At this point, the input noisy image information gets segregated into object and background regions thereby leading to extracted object centric features from the input noisy image scene.

B. Limitations of MLSONN architecture

The MLSONN architecture uses a sigmoidal function based activation with single point fixed and uniform thresholding. Therefore, it assumes homogeneity of the input image information. However, in real world situations, images exhibit a fair amount of heterogeneity in its information content which encompasses over the entire image pixel neighborhoods. This limitation of the MLSONN architecture is evident from the

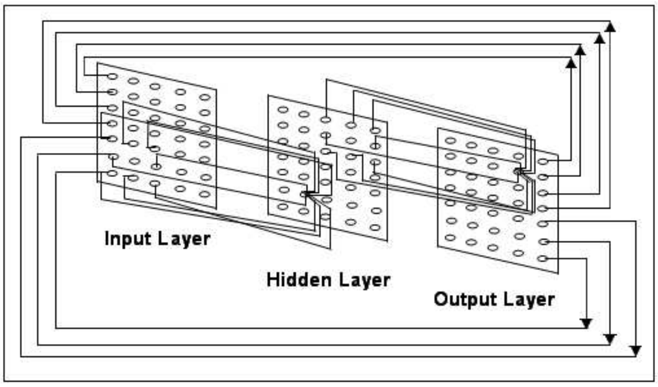

Fig. 1. Schematic of a three layer MLSONN architecture employing second order neighborhood topology based interconnections

fact that the network structure employed does not include any provision for accommodating the image context information for determining the activation function thresholds. This problem of fixed and uniform thresholding can be better enunciated if a standard beta activation function is used as an activation function for the network operation.

The standard beta function with single and fixed point thresholding is given by

$$f(t) = \int_0^t Kx^{\alpha}(1-x)^{\beta}dx, \quad \alpha, \beta \geq 0, \quad t \in [0,1] \tag{10}$$

where, t represents the class widths and K is a normalizing constant such that,

$$K = \frac{1}{\int_0^1 x^{\alpha}(1-x)^{\beta}dx} \tag{11}$$

The α and β parameters control the shape and slope of the beta function respectively. The operating point of the beta function is determined by the threshold value of the function. The threshold value of the beta function is given as

$$\tau = \frac{\alpha}{\alpha+\beta} \tag{12}$$

From the expression of the threshold parameter (τ), it is evident that both the α and β parameters play a vital role in determining the transfer characteristics of the beta activation function. The performance of the activation function i.e. the performance of the MLSONN architecture largely depends on the choice of the α and β parameters. This can be substantiated by observing the MLSONN performance using different sets of the α and β parameters.

C. Binary object extraction by MLSONN architecture using different α and β parameters

A study of the performance or extraction capabilities of a MLSONN architecture using the standard beta activation function has been carried out with different sets of α and β parameters. Experiments have been conducted on noisy versions of an image affected with different Gaussian noise levels of zero mean and standard deviation of σ=8, 10, 12, 14 and 16, using three representative values of the α parameter viz. {α=0.25, 0.5, 0.75}. The corresponding sets of the β parameters chosen are given in Table I.

TABLE I
SELECTED β VALUES FOR DIFFERENT α

α	β
0.25	{0.05, 0.1, 0.15, 0.2, 0.25, 0.3}
0.5	{0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6}
0.75	{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8}

1) Extraction efficiency of MLSONN architecture: As a measure of the extraction efficiency of the MLSONN architecture, Ghosh *et al.* [23] proposed a percentage of correct classification of pixels (pcc) which is defined as

$$pcc = \frac{tnocc \times 100}{tnop} \tag{13}$$

where $tnocc$ is the total number of pixels correctly classified into object/background pixels and $tnop$ is the total number of pixels in the image scene.

The pcc values obtained for the aforementioned different sets of β parameters for α={0.25, 0.5, 0.75} and σ=8, 10, 12, 14, and 16 are listed in Tables II, III and IV respectively. Variations of the pcc values with the β parameter for α={0.25, 0.5, 0.75} are shown in Fig. 2, 3 and 4 respectively.

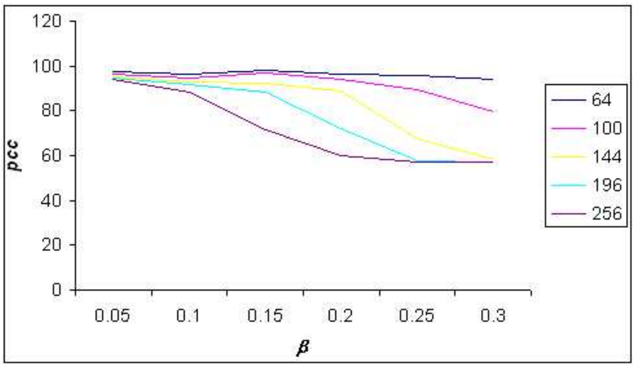

Fig. 2. Variation of pcc with β at α=0.25

TABLE II

pcc VALUES FOR α=0.25 AND DIFFERENT NOISE LEVELS

σ	β					
	0.05	0.1	0.15	0.2	0.25	0.3
8	97.48	96.53	97.83	96.15	95.91	94.03
10	96.12	94.78	96.90	94.23	89.68	79.69
12	94.93	93.11	92.42	89.07	67.31	58.48
14	94.55	91.66	88.42	72.12	57.52	57.37
16	93.90	88.47	71.38	59.98	57.32	57.37

TABLE III

pcc VALUES FOR α=0.5 AND DIFFERENT NOISE LEVELS

σ	β											
	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6
8	96.56	96.69	96.86	96.84	99.07	99.66	99.11	98.24	97.95	97.89	97.94	96.21
10	95.07	96.23	96.69	96.77	96.45	98.24	98.50	97.79	97.61	97.12	96.34	95.08
12	91.29	94.96	96.01	96.17	95.51	94.89	96.58	95.77	94.86	92.14	91.25	84.74
14	83.59	90.75	94.31	95.53	95.37	94.250	94.82	93.378	91.81	84.99	75.88	68.41
16	63.01	80.77	90.64	94.48	94.77	92.63	92.41	86.03	80.82	68.93	65.38	62.15

TABLE IV

pcc VALUES FOR α=0.75 AND DIFFERENT NOISE LEVELS

σ	β							
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
8	94.97	96.46	96.86	97.26	99.75	98.85	97.96	97.94
10	90.62	94.99	96.55	96.90	98.69	97.88	97.797	95.89
12	84.55	91.27	95.96	96.22	96.39	97.15	94.77	92.31
14	69.12	86.29	93.58	95.67	95.22	94.51	91.46	83.80
16	47.35	71.16	88.59	94.94	94.15	92.64	81.55	66.35

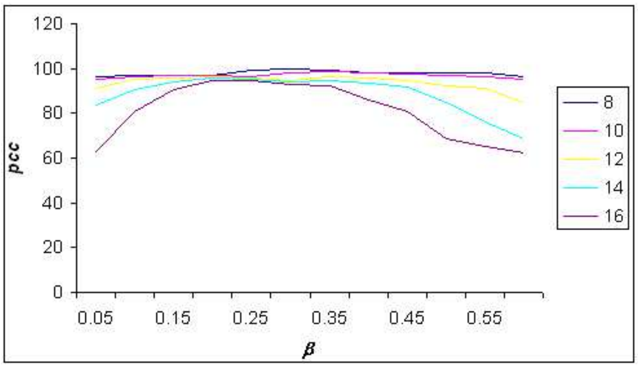

Fig. 3. Variation of *pcc* with β at α=0.5

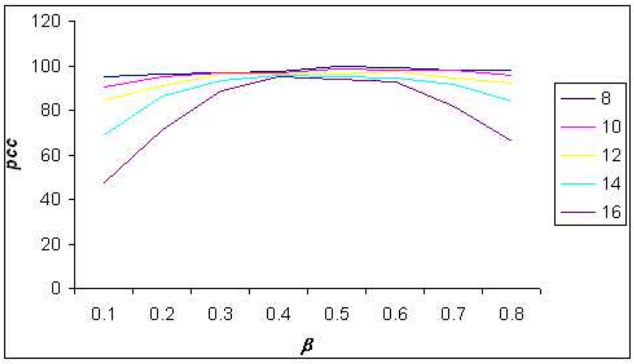

Fig. 4. Variation of *pcc* with β at α=0.75

From Fig. 2, it is seen that the maximum *pcc* values are obtained with β=0.05 irrespective of the noise levels, whereas it is seen from Fig. 3 and 4 that the maximum *pcc* values of 94.77 and 94.94 for the highest noise level of σ=16 are obtained with β=0.25 and 0.4 respectively. Thus the best possible combinations of $\{\alpha, \beta\}$ parameters for the selected α parameters, which would result in the best qualities of the extracted images within the selected range of α values, are $\{0.25, 0.05\}$, $\{0.5, 0.25\}$ and $\{0.75, 0.4\}$.
Extraction of binary objects has been carried out on a synthetic image and a real life spanner image (shown in Fig. 5) employing these selected parameters. The noisy versions of the images used are shown in Fig. 6. The corresponding extracted images pertaining to $\{\alpha, \beta\}$=[$\{0.25, 0.05\}$, $\{0.5, 0.25\}$, $\{0.75, 0.4\}$] for σ=8, 10, 12, 14 and 16 are shown in Fig. 7 and 8 respectively.

2) Time efficiency of MLSONN architecture: The MLSONN architecture resorts to backpropagation based weight adjustment techniques, which involve time complex computational overhead. The operational times of the MLSONN architecture for the extraction of the noisy synthetic and real life spanner image for the different $\{\alpha, \beta\}$ parameters as reported in Table I, are shown in Tables V, VI and VII.

From the results of object extraction obtained using a

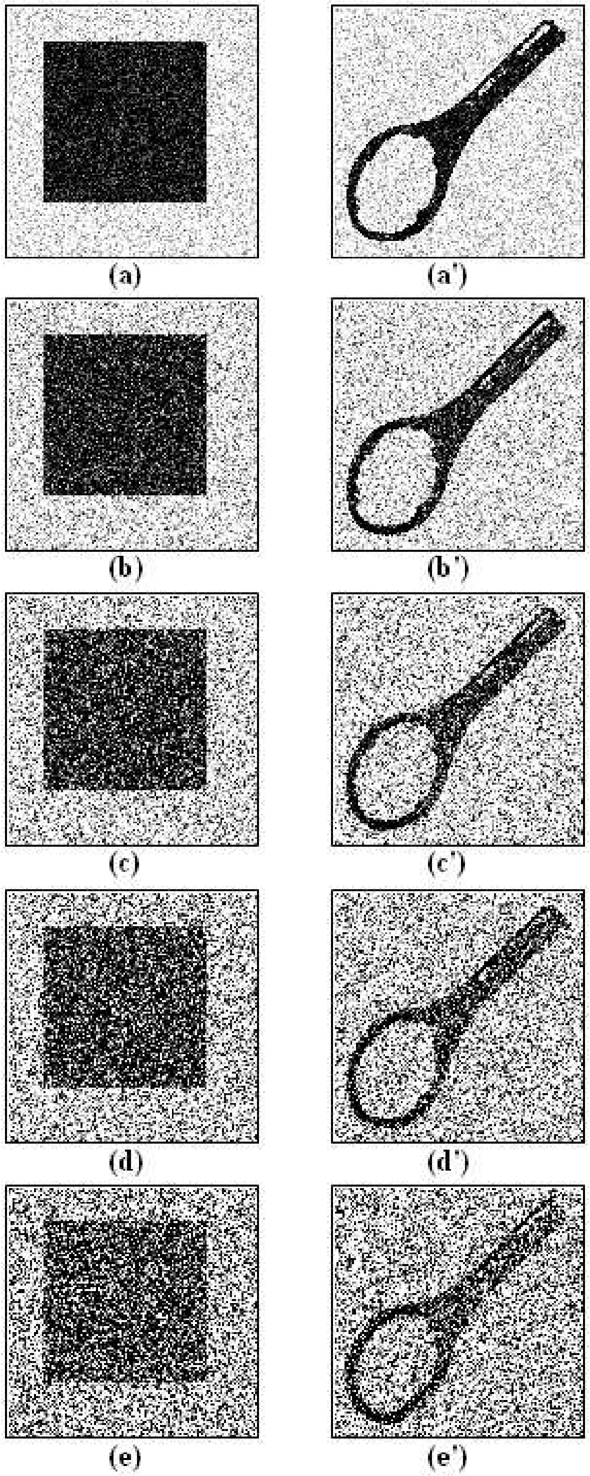

Fig. 6. Noisy images (a)(b)(c)(d)(e) synthetic images (a′)(b′)(c′)(d′)(e′) spanner images at σ=8, 10, 12, 14 and 16

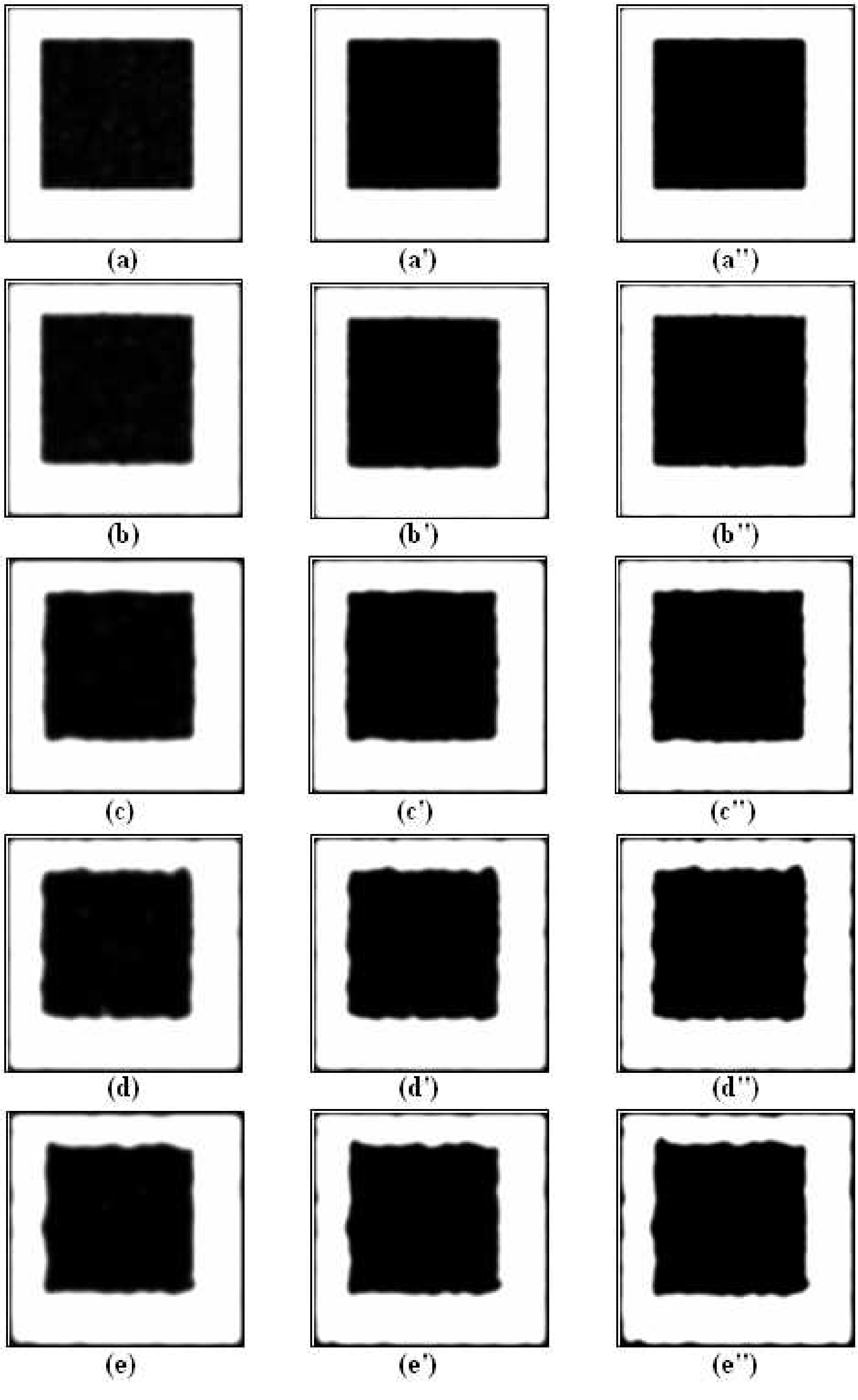

Fig. 7. Extracted synthetic images for σ=8, 10, 12, 14 and 16 (a)(b)(c)(d)(e) at $\{\alpha, \beta\}$= {0.25, 0.05} (a′)(b′)(c′)(d′)(e′) at $\{\alpha, \beta\}$={0.5, 0.25} (a′′)(b′′)(c′′)(d′′)(e′′) at $\{\alpha, \beta\}$={0.75, 0.4}

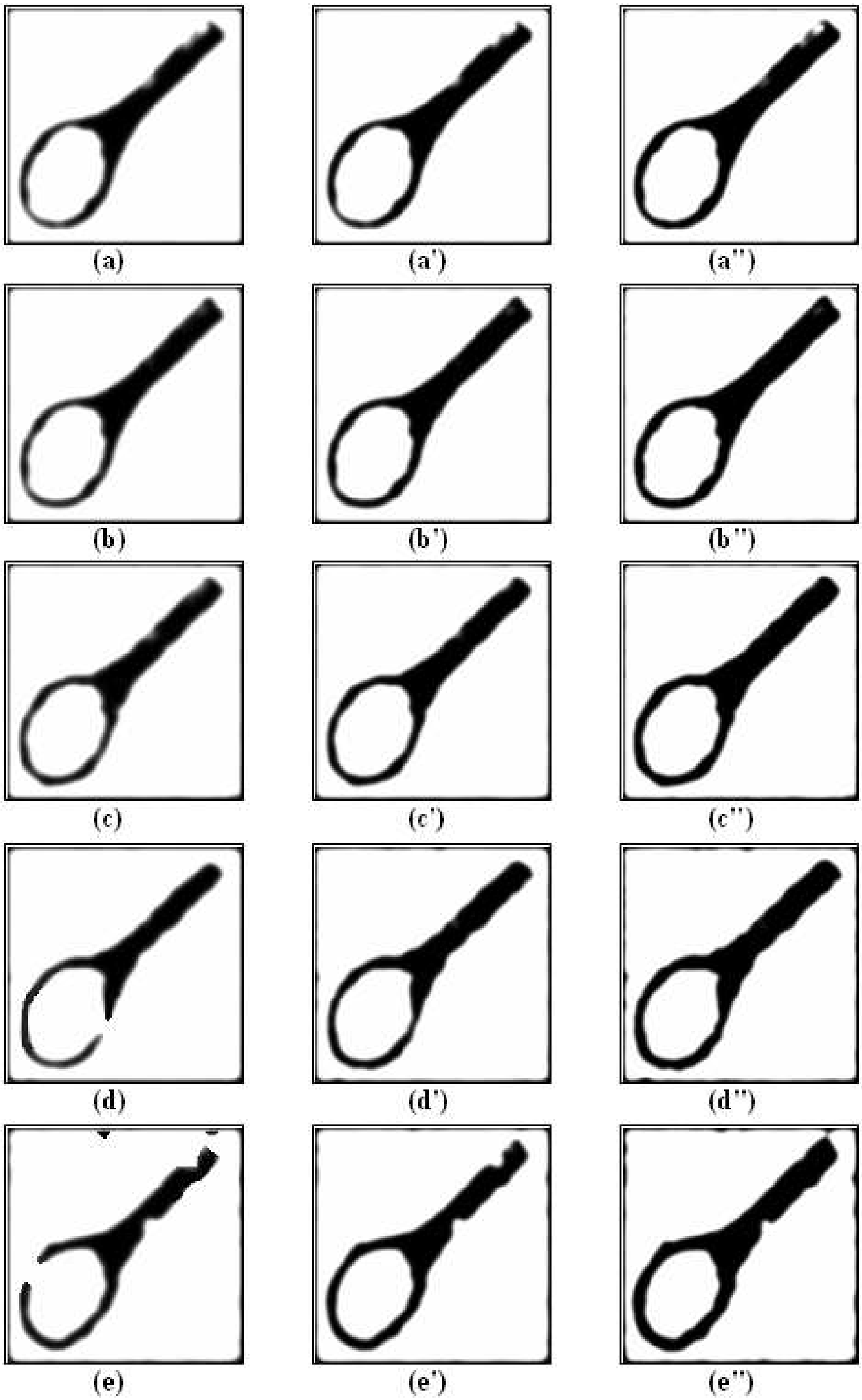

Fig. 8. Extracted spanner images for σ=8, 10, 12, 14 and 16 (a)(b)(c)(d)(e) at $\{\alpha, \beta\}$={0.25, 0.05} (a′)(b′)(c′)(d′)(e′) at $\{\alpha, \beta\}$={0.5, 0.25} (a′′)(b′′)(c′′)(d′′)(e′′) at $\{\alpha, \beta\}$={0.75, 0.4}

TABLE V
EXTRACTION TIME IN SECONDS FOR α=0.25 AND DIFFERENT NOISE LEVELS

σ	β					
	0.05	0.1	0.15	0.2	0.25	0.3
8	156	233	232	311	313	740
10	236	313	314	473	801	753
12	272	474	719	656	905	816
14	471	638	984	781	1020	1246
16	625	968	1130	820	1040	–

TABLE VI
EXTRACTION TIME IN SECONDS FOR α=0.5 AND DIFFERENT NOISE LEVELS

σ	β											
	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6
8	152	230	152	154	153	154	206	231	231	153	153	230
10	229	231	228	154	230	154	273	253	273	232	232	312
12	252	307	302	232	308	310	311	333	394	414	322	412
14	566	447	373	307	383	389	471	556	403	477	422	453
16	570	701	508	452	457	698	556	576	418	526	440	448

TABLE VII
EXTRACTION TIME IN SECONDS FOR α=0.75 AND DIFFERENT NOISE LEVELS

σ	β							
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
8	150	150	150	150	151	151	151	151
10	296	223	225	151	152	217	219	246
12	400	368	225	228	219	305	465	468
14	431	396	368	301	304	616	571	485
16	448	426	427	445	533	627	640	606

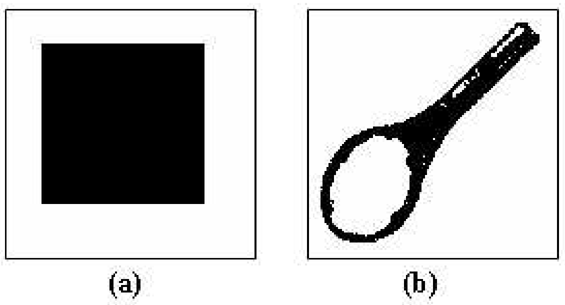

Fig. 5. Original images (a) synthetic image (b) spanner image

MLSONN architecture guided by a beta activation function, it is clear that several choices of the β parameter are possible for a given α which would produce better quality extracted images. This is due to the inherent heterogeneity of image information content, which remains unattended by the MLSONN architecture. Moreover, the time complexity of the object extraction approach with the MLSONN architecture is also evident from Tables V, VI and VII. These values are partly due to the time complex backpropagation algorithm employed in the interconnection weight adjustment procedure. Moreover, it is found that the entire image region gets wiped out for $\sigma = 16$ and $\{\alpha, \beta\}$={0.25, 0.3} after a certain number of iterations. This is indicated by the corresponding entry in Table V.

It may however, be noted that the MLSONN architecture has been efficiently applied for several image processing and segmentation applications including tracking of multiple moving targets from a video scene [26]. The MLSONN architecture has been modified to a layered version for the purpose of multiscale object extraction [27]. However, the introduction of different layers to the architectural model increases network complexity as well. Bhattacharyya *et al.* applied a fuzzy cardinality estimate based approximation of input multiscale image scenes for extracting multiscale objects by the original MLSONN architecture. Multiscaling has also been induced in the MLSONN architecture by resorting to a functional modification of the architecture keeping the architectural structure unchanged [29][30][31][32][33][34]. On similar lines, an extended parallel version (PSONN) of three different MLSONN architectures has been efficiently employed for the extraction of pure and true color objects from a color image scene [35][36][37][38][39]. Each of these approaches suffers from the computational overhead imposed by the underlying backpropagation based weight adjustment procedure. Bhattacharyya *et al.* [40][41] devised a pruning algorithm for evolving a refined MLSONN architecture with reduced number of interconnections between the different layers of the network architecture to enhance the time efficiency of the extraction procedure.

IV. SELF SUPERVISED BI-DIRECTIONAL NEURAL NETWORK (BDSONN) ARCHITECTURE

It is already stated that the MLSONN architecture suffers from some serious limitations in that it lacks the power of

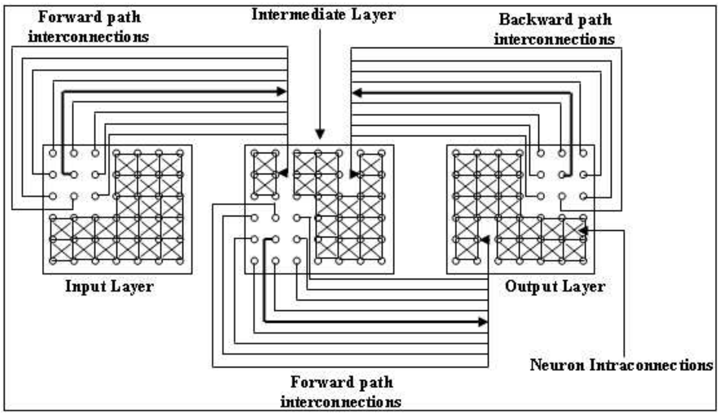

Fig. 9. Schematic layout of the Bi-directional self-organizing neural network (BDSONN) architecture using a second order neighborhood based forward and backward path *inter*-layer interconnections for the propagation of network states [bold lines indicate path for propagation of fuzzy context sensitive thresholding information, not all *intra*-layer interconnections are shown for the sake of clarity]

adaptation to the image context as well as the use of the standard backpropagation algorithm impedes its use in real time applications. In this section, a new bi-directional self organizing neural network architecture is presented which attempts to solve the problems encountered by the original MLSONN architecture.

The proposed self supervised bi-directional three layer neural network (BDSONN) architecture is a fully connected neural network architecture. It comprises an input layer, an intermediate layer and an output layer of neurons. The number of neurons in each of the network layers corresponds to the number of pixels in the input image scene. The fuzzy membership values of the input image scene are fed as input to the input layer. The neurons in each layer of the network are connected to each other within the same layer with full and fixed *intra*-layer interconnection strengths. Each neuron in a particular layer of the network is connected to the second order neighbors of the corresponding neuron in the previous layer following a second order neighborhood-based topology. A schematic layout of the network architecture is shown in Fig. 9. The *inter*-layer connection strengths are decided by the relative fuzzy membership values of the neighbors and the candidate neuron and are thereby influenced by the local heterogeneity within the neighborhood fuzzy subsets in the image scene. If μ_{kj} is the membership value at the j^{th} candidate neuron in the k^{th} layer and μ_{kj} is the membership value at its i^{th} second order neighbor in the same layer, then the *inter*-layer connection strength, w_{kilj}, between the corresponding candidate neuron of the next l^{th} layer and the i^{th} second order neighbors of the k^{th} layer is given by

$$w_{kilj} = \overline{\mu_{kj} - \mu_{ki}} \tag{14}$$

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 i^{th} 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 neighborhood, is given as

$$I_{lj} = \sum_i w_{kilj} I_{ki} \tag{15}$$

where, w_{kilj} are the *inter*-layer interconnection weights. The output, O_j, produced by this neuron is given by

$$O_j = f(I_{lj}) \tag{16}$$

where, f is the beta activation function with context sensitive thresholding and is given as

$$f(t) = \int_0^t K x^{\alpha}(1-x)^{\beta_C} dx, \quad \alpha, \beta_C \geq 0, \quad t \in [0,1] \tag{17}$$

where, t and K have their usual significances. The β_C 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 estimates of the neighborhood fuzzy subsets in the form of the β_C parameter is given by

$$\tau_c = \frac{\alpha}{\alpha + \beta_C} \tag{18}$$

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 fixed 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 *fro* 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.

A. Network self-organization algorithm

The self-supervised operation of the proposed bi-directional self-organizing neural network (BDSONN) architecture 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 fuzzified image pixel information at row *m* and column *n* at the l^{th} 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 l^{th} and $(l+1)^{th}$ network layers at a particular 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
```

B. Stabilization of the network

The principle of the object extraction process is the localization of object centric features from a noisy background. The fuzziness in a noisy image scene comprising object information and background information is due to the induced noise therein. The presence of noise results in a fair amount of heterogeneity (as regards to the intensity levels) between the individual image pixels in the different neighborhood fuzzy subsets in the image scene. This heterogeneity manifested in the neighborhood fuzzy hostilities is an indirect measure of the degree of noise in the image scene. The network system error (ψ) is thus a function of the neighborhood fuzzy hostility index (ζ) and can be represented as

$$\psi = f(\zeta) = f(\frac{3}{8}\sum_{i=1}^{8}\frac{|\mu_p - \mu_{q_i}|}{|\mu_p + 1| + |\mu_{q_i} + 1|}) \qquad (19)$$

Recalling equation 14, ψ is thus a function of w_{ij}, the *inter*-layer interconnection weights between the i^{th} and the j^{th} layer neurons. Thus, ψ assumes minimum value when ζ is minimum, i.e. when $\mu_p - \mu_{q_i}$=0. This implies a fuzzy hostility index value of ζ=0. Thus, it can be inferred that the system attains stabilization, when the heterogeneity of the system is at a minimum.

In addition, the fuzzy entropy measures of these fuzzy hostilities also reflect of the average amount of ambiguity in the image scene. The task of object extraction is tantamount to the reduction in the fuzzy hostilities in the image neighborhood regions or localizing homogeneous object and background regions out of the heterogeneous noisy regions, thereby reducing the average fuzzy entropy measures of the fuzzy neighborhood hostilities. This implies that in an extracted image scene comprising of homogeneous object and background regions, the fuzzy entropy measures of the fuzzy neighborhood hostilities are minimum and have attained stabilization. Therefore, the convergence of the network operation is determined by the stability achieved in the fuzzy entropy measures. Thus the stabilization of the object extraction process by the network is decided by the stabilization of the fuzzy entropy measures.

V. Results of object extraction

The proposed BDSONN architecture has been applied for the extraction of binary objects from a noisy background. One synthetic image and one real life spanner image (Fig. 5), corrupted with Gaussian noise of zero mean and standard deviation σ=8, 10, 12, 14 and 16 (Fig. 6) are used as the input images to the network architecture for extraction of object centric regions from the noisy backgrounds. Experiments have been conducted with α={0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, 1.0}. The adaptive β parameters for the beta activation function employed in the extraction process, are determined dynamically from the image context through the estimation of the fuzzy cardinality estimates of the different neighborhood fuzzy subsets of the input noisy image scenes. The extracted versions of the synthetic and spanner images for different noise levels and with the different α parameters are shown in Fig. 10, 11, 12 and 13.

A. Performance of the proposed architecture

Considering the proposed network architecture to be a noise immune system, where noisy inputs are converted into non-noisy versions, the performance of the system can be evaluated by a system transfer index (nif), which reflects the noise immunity of the system. It can be mathematically expressed as

$$nif = \rho(O, N) \times \rho(N, E) \tag{20}$$

where, $\rho(O, N)$ refers to the ratio of the coefficients of variation of the neighborhood fuzzy subset hostility indices in the original and the noisy images and $\rho(N, E)$ refers to the ratio of the coefficients of variation of the neighborhood fuzzy subset hostility indices in the noisy and the extracted counterparts. From the definition of the system transfer index, it is clear that closer the value of the index is to unity, the better extraction it reveals. Table VIII show the noise immunity factor (nif) for the two images for different levels of noises and α={0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, 1.0}.

From the table it is evident that the network architecture is efficient in retrieving object regions from different degrees of noisy environment. Moreover, the nif values show that the performance of the network as regards to object extraction degrades at higher noise levels, which is obvious from the fact that higher noise levels means greater probability of an object pixel being surrounded by noise pixels and hence greater chances of misclassification of an image pixel into a background pixel than an object pixel. This is also represented in Fig. 14 and 15, which show the variation of the noise immunity factors (nif) of the proposed BDSONN architecture with the α parameter at different noise levels for the two test images.

TABLE VIII

Noise immunity factors (nif) for α={0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, 1.0} at different noise levels

α	σ	Synthetic image	Spanner image
0.125	8	1.4339	1.2083
	10	1.5721	1.2601
	12	1.6095	1.2855
	14	1.6532	1.2870
	16	1.6704	1.3069
0.25	8	1.2392	1.0663
	10	1.4026	1.1471
	12	1.4468	1.1910
	14	1.5413	1.2352
	16	1.6040	1.2478
0.375	8	1.0211	0.9818
	10	1.2503	1.0489
	12	1.3982	1.1624
	14	1.49375	1.2064
	16	1.55849	1.2409
0.5	8	1.0055	0.9624
	10	1.1392	0.9950
	12	1.3401	1.0892
	14	1.4684	1.1615
	16	1.5118	1.2344
0.625	8	1.0035	0.9717
	10	1.0976	0.9941
	12	1.2727	1.0492
	14	1.4249	1.1434
	16	1.5440	1.2303
0.75	8	1.0012	0.9721
	10	1.06810	0.9909
	12	1.2244	1.0315
	14	1.4157	1.1188
	16	1.5329	1.2164
0.875	8	1.0008	0.9753
	10	1.0605	0.9898
	12	1.1818	1.0267
	14	1.3808	1.0951
	16	1.5477	1.2036
1.0	8	1.0008	0.9765
	10	1.0407	0.9887
	12	1.1594	1.0236
	14	1.3571	1.0874
	16	1.5184	1.1877

In addition to the proposed system transfer index which is a measure of the noise immunity factor (nif) of the proposed BDSONN architecture, the pcc[23] values for the extracted images with α={0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, 1.0} are also computed using equation 12. The computed pcc values, which reflect the quality of the extracted images for σ=8, 10, 12, 14 and 16 are shown in Table IX. Fig. 16 and 17 show the variation of the computed pcc values for the two images with different values of the α parameter at different noise levels respectively.

Fig. 16 and 17 show that the pcc values exhibit an increasing trend as α increases until the maximum pcc values are obtained at α=1.0.

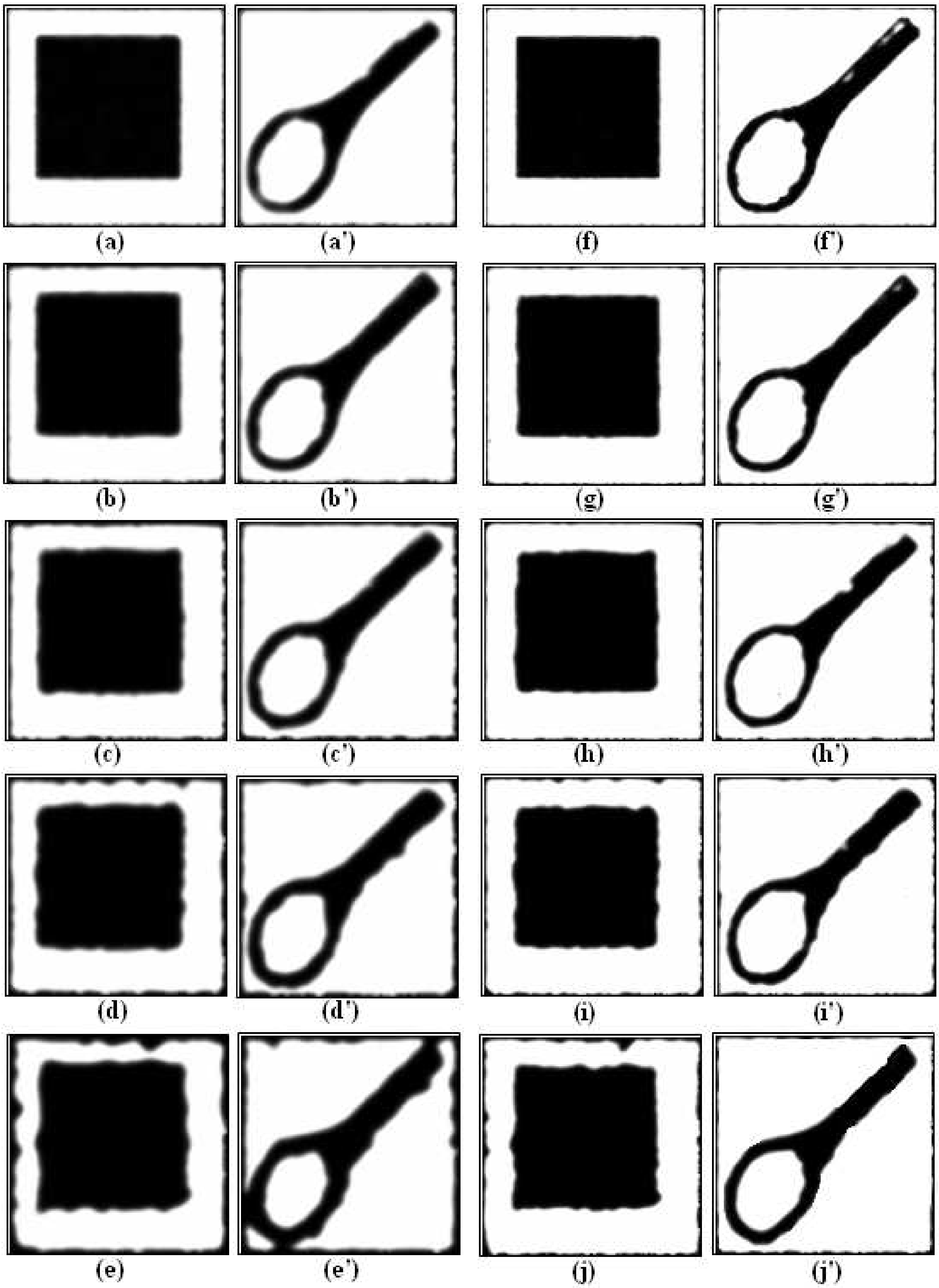

Fig. 10. Extracted images for σ=8, 10, 12, 14 and 16 (a)(b)(c)(d)(e) synthetic images at α=0.125 (a′)(b′)(c′)(d′)(e′) spanner images at α=0.125 (f)(g)(h)(i)(j) synthetic images at α=0.25 (f′)(g′)(h′)(i′)(j′) spanner images at α=0.25

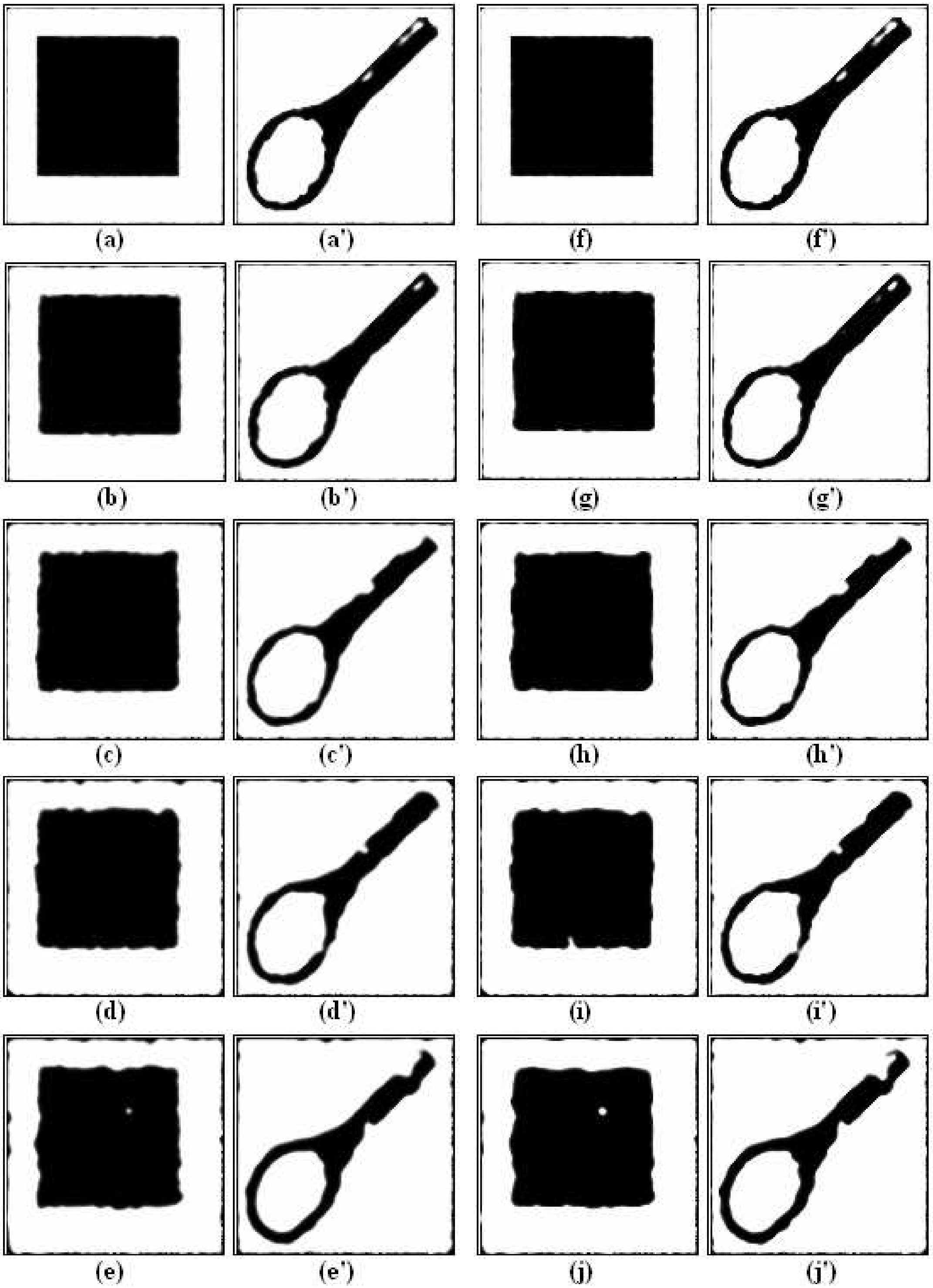

Fig. 11. Extracted images for σ=8, 10, 12, 14 and 16 (a)(b)(c)(d)(e) synthetic images at α=0.375 (a′)(b′)(c′)(d′)(e′) spanner images at α=0.375 (f)(g)(h)(i)(j) synthetic images at α=0.5 (f′)(g′)(h′)(i′)(j′) spanner images at α=0.5

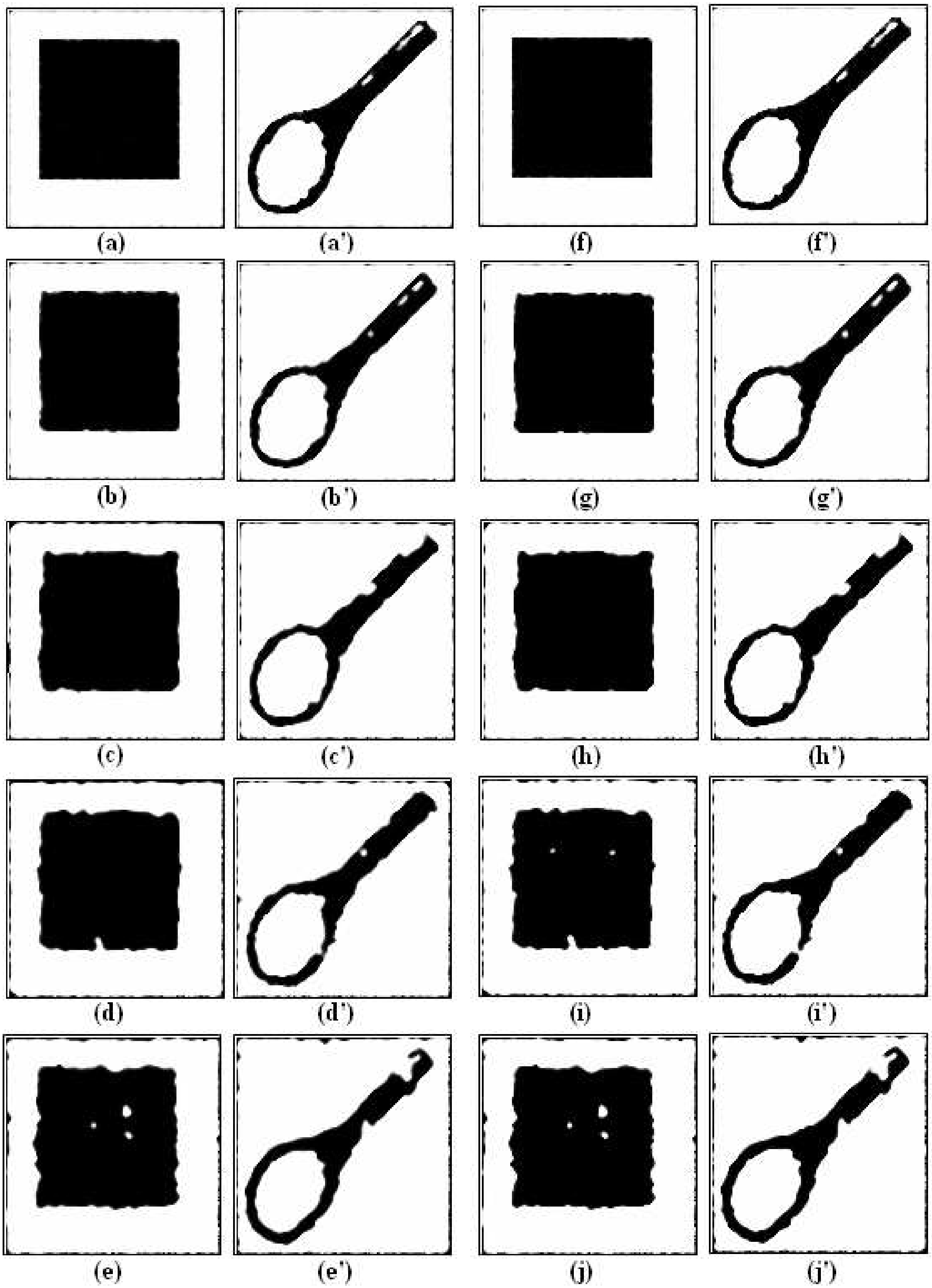

Fig. 12. Extracted images for σ=8, 10, 12, 14 and 16 (a)(b)(c)(d)(e) synthetic images at α=0.625 (a′)(b′)(c′)(d′)(e′) spanner images at α=0.625 (f)(g)(h)(i)(j) synthetic images at α=0.75 (f′)(g′)(h′)(i′)(j′) spanner images at α=0.75

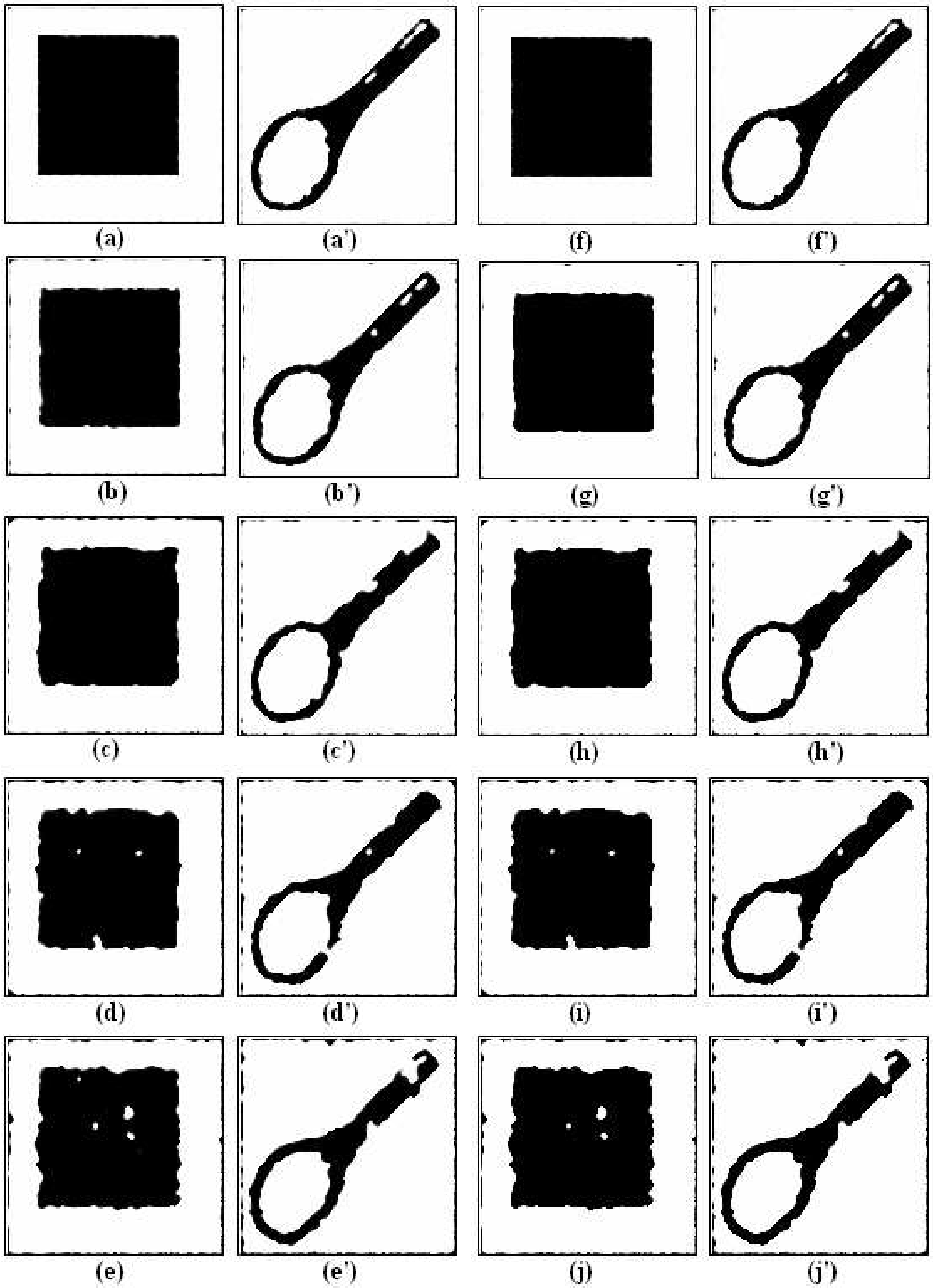

Fig. 13. Extracted images for σ=8, 10, 12, 14 and 16 (a)(b)(c)(d)(e) synthetic images at α=0.875 (a′)(b′)(c′)(d′)(e′) spanner images at α=0.875 (f)(g)(h)(i)(j) synthetic images at α=1.0 (f′)(g′)(h′)(i′)(j′) spanner images at α=1.0

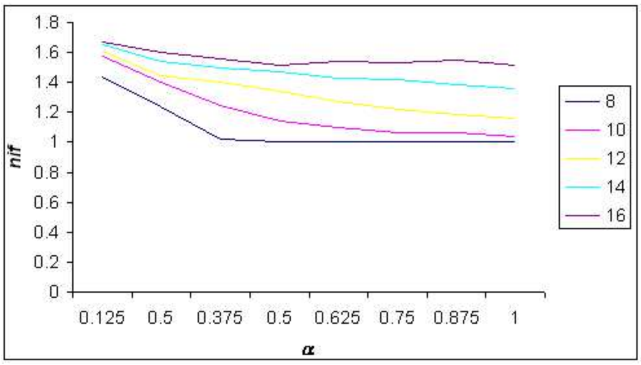

Fig. 14. Variation of nif with α for σ=8, 10, 12, 14 and 16 during the extraction of the synthetic image

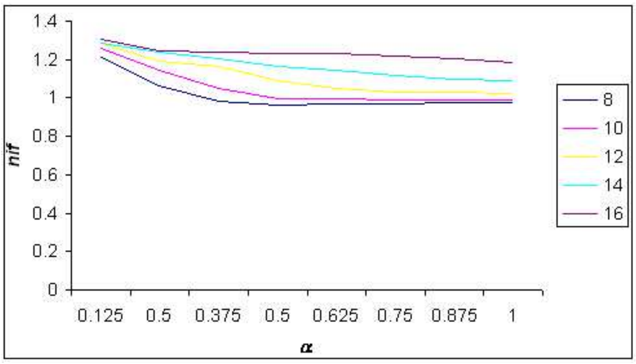

Fig. 15. Variation of nif with α for σ=8, 10, 12, 14 and 16 during the extraction of the spanner image

1) Time efficiency of the proposed BDSONN architecture: The BDSONN architecture carries out the act of self supervision by means of bi-directional propagation of network states between the network intermediate and output layers. This implies that the network intermediate and output layers act as competitive layers for retrieving the object centric regions out of the noisy environment. Moreover, the proposed architecture does not resort to the time complex backpropagation based weight adjustment procedure. Instead, the adjustment and reassignment of the adjusted weights are carried out in a deterministic fashion through the relative strengths of the fuzzy membership values at the participating neurons of the different layers of the network, thereby reducing the computational burden, which would otherwise be imposed upon if the standard backpropagation algorithm is used in the adjustment process. The extraction time required by the proposed network architecture for the extraction of the two images from different noisy environments with the chosen values of the α parameter are shown in Tables X and XI respectively. From the tables it is evident that the proposed BDSONN network operates much faster compared to the MLSONN architecture as regards to extraction of objects from a noisy background.

Fig. 18 to 23 show the variation of the coefficient of variation of the fuzzy hostility indices (FHI) of the noisy synthetic and spanner images respectively along with the BDSONN extracted images during the object extraction process with respect to the number of iterations required during the

TABLE IX

pcc VALUES FOR α=$\{0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, 1.0\}$ AT DIFFERENT NOISE LEVELS

α	σ	Synthetic image	Spanner image
0.125	8	96.149	93.14
	10	94.80	92.77
	12	93.54	91.18
	14	90.53	88.68
	16	86.17	82.95
0.25	8	98.91	97.90
	10	97.69	96.46
	12	96.72	95.29
	14	95.65	94.39
	16	94.04	93.40
0.375	8	99.87	99.03
	10	98.52	97.66
	12	97.07	95.60
	14	96.06	95.05
	16	95.28	93.65
0.5	8	99.93	99.27
	10	98.80	97.94
	12	96.93	96.26
	14	95.74	95.48
	16	95.42	93.96
0.625	8	99.94	100.00
	10	99.07	98.47
	12	97.41	97.01
	14	96.17	96.16
	16	95.62	94.46
0.75	8	99.94	99.94
	10	99.05	98.56
	12	97.61	97.14
	14	96.78	92.79
	16	95.68	94.63
0.875	8	99.94	99.89
	10	99.05	98.54
	12	97.61	97.21
	14	96.34	96.44
	16	95.68	94.82
1.0	8	99.94	99.86
	10	99.14	98.53
	12	97.85	97.29
	14	96.66	96.57
	16	95.73	94.94

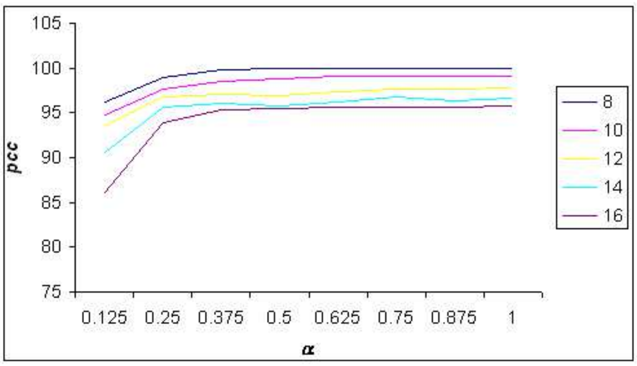

Fig. 16. Variation of pcc with α at σ=8, 10, 12, 14 and 16 for the synthetic image

TABLE X

EXTRACTION TIME IN SECONDS FOR SYNTHETIC IMAGE FOR α={0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, 1.0} AT DIFFERENT NOISE LEVELS

σ	α							
	0.125	0.25	0.375	0.5	0.625	0.75	0.875	1.0
8	144	109	108	109	72	72	72	71
10	210	178	174	185	142	179	106	216
12	236	236	202	251	211	206	205	243
14	292	245	203	281	235	212	236	249
16	373	261	268	420	242	273	248	321

TABLE XI

EXTRACTION TIME IN SECONDS FOR SPANNER IMAGE FOR α={0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, 1.0} AT DIFFERENT NOISE LEVELS

σ	α							
	0.125	0.25	0.375	0.5	0.625	0.75	0.875	1.0
8	294	129	170	131	86	86	145	85
10	324	213	249	216	170	171	171	172
12	331	280	285	289	200	245	254	255
14	346	293	297	300	254	254	279	289
16	442	310	362	324	288	280	289	367

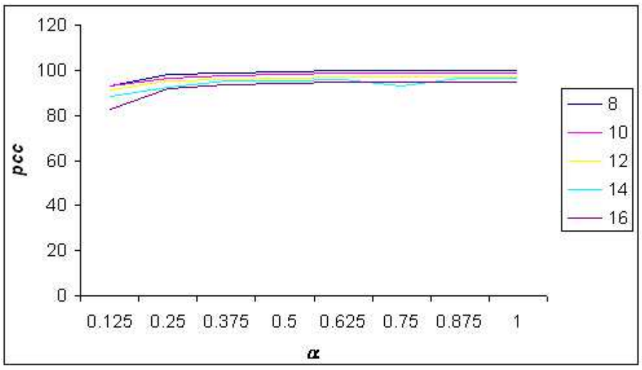

Fig. 17. Variation of pcc with α at σ=8, 10, 12, 14 and 16 for the spanner image

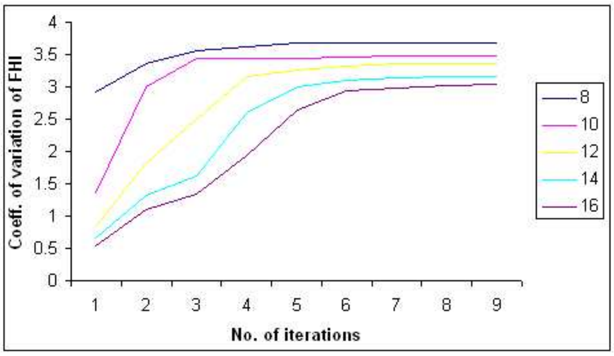

Fig. 18. Variation of neighborhood fuzzy hostility indices of synthetic image with time at α=0.25

extraction process for α={0.25, 0.5, 0.75}.

Fig. 24 to 29 indicate the corresponding variations in the fuzzy entropy measures of the fuzzy hostility indices during the extraction process for the two images with the selected α parameter values.

VI. CONCLUSIONS

The multilayer self organizing neural network (MLSONN) architecture is described. The limitations of the network architecture in terms of the thresholding methodology and the error adjustment methodology are discussed and addressed.

A bi-directional self organizing neural network (BDSONN) architecture with self supervising features, suitable for object extraction from a noisy background and capable of removing the limitations of the MLSONN architecture, has been proposed. The architecture comprises three layers of fully connected neurons. The second and the third layer of neurons counter-propagate the network states and in the process achieve a self supervised organization of the network inputs fed at the input layer. The network uses a beta activation function aided by an adaptive image context sensitive threshold values which are determined from the fuzzy cardinality estimates of the pixel neighborhood fuzzy subsets of the image scene. The choice of the adaptive image context sensitive threshold values for the functional operation of the network enhances the generalization capabilities of the network architecture as the network takes into cognizance the inherent heterogeneity in the input image scene. The assignment and updating of interconnection weights are carried out by evaluating the relative measures of the fuzzy membership values of the neuron states, thereby reducing the time complexity of the object extraction procedure.

Results of application of the proposed architecture are demonstrated with two images affected with various degrees of Gaussian noise. The extraction efficiency of the proposed architecture is also determined by a proposed noise immunity factor. Extraction times reported indicate faster convergence of the network parameters during the object extraction procedure as compared to the MLSONN architecture for the same degrees of noise. It is also observed that the proposed architecture maintains the shapes and boundaries of the images after extraction. However, it remains to investigate the performance of the architecture in the extraction of multiscale gray and color images. The authors are currently engaged in this direction.

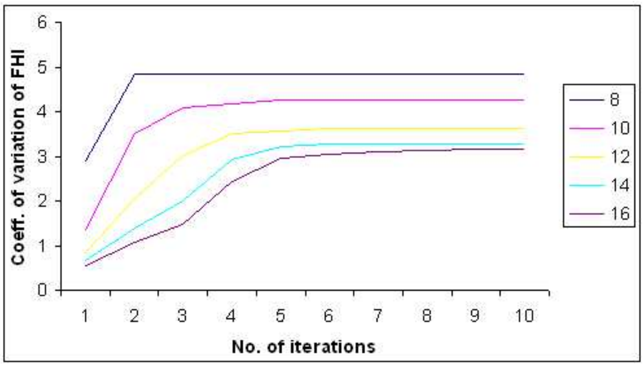

Fig. 19. Variation of neighborhood fuzzy hostility indices of synthetic image with time at α=0.5

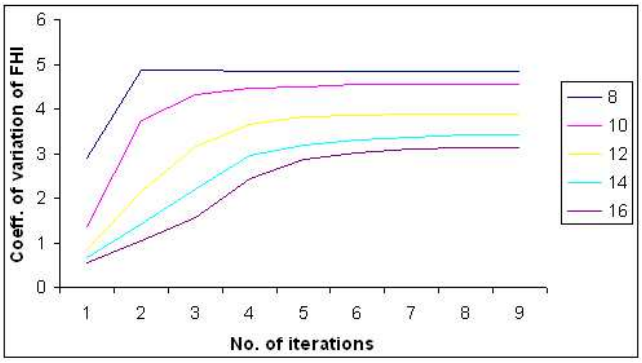

Fig. 20. Variation of neighborhood fuzzy hostility indices of synthetic image with time at α=0.75

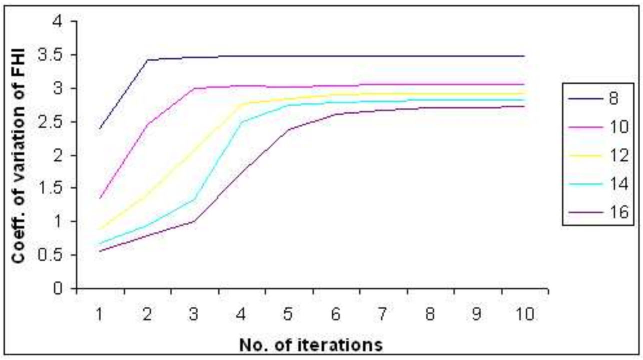

Fig. 21. Variation of neighborhood fuzzy hostility indices of spanner image with time at α=0.25

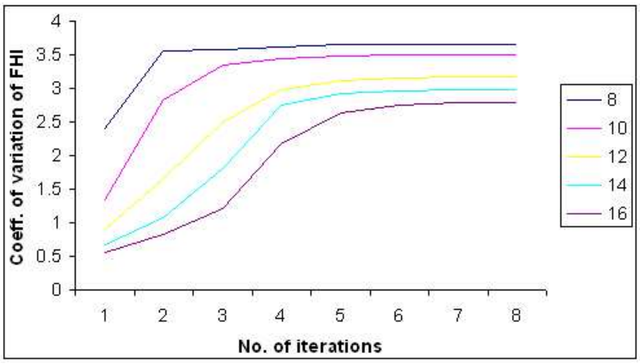

Fig. 22. Variation of neighborhood fuzzy hostility indices of spanner image with time at α=0.5

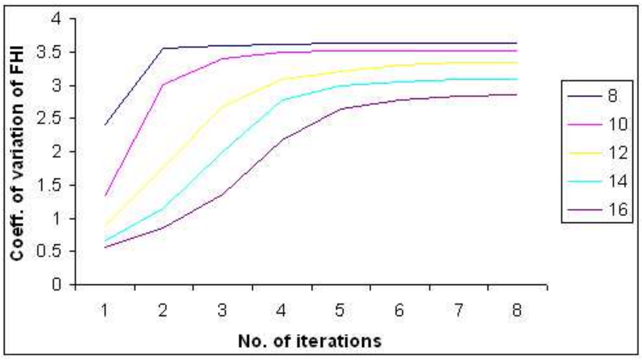

Fig. 23. Variation of neighborhood fuzzy hostility indices of spanner image with time at α=0.75

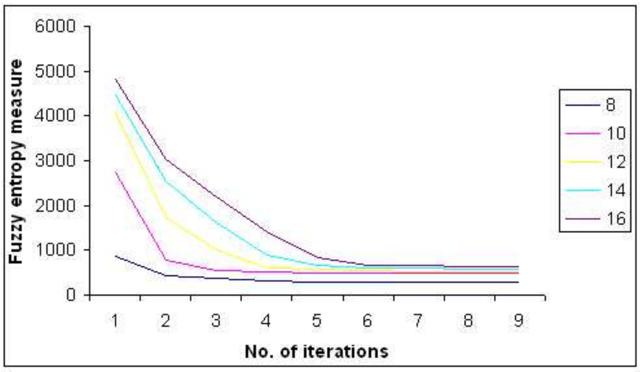

Fig. 24. Variation of fuzzy entropy measures of neighborhood fuzzy hostility indices of synthetic image with time at α=0.25

REFERENCES

[1] R. C. Gonzalez and P. Wintz, *Digital Image Processing*, MA: Addison-Wesley, 1977.
[2] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*, vol. 1, 2nd eds., New York: Academic Press, 1982.
[3] M. P. Ekstrom, ed. *Digital Image Processing Techniques*, New York: Academic Press, 1984.
[4] M. R. Banham and A. K. Katsaggelos, *Digital image restoration*, IEEE Signal Processing Magazine, vol. 14, no. 2, pp. 24-41, 1997.
[5] M. Egmont-Petersen and T. Arts, "Recognition of radiopaque markers in X-ray images using a neural network as nonlinear filter," *Pattern Recognition Letters*, vol. 20, no. 5, pp. 521-533, 1999.
[6] S. Haykin, *Neural networks: a comprehensive foundation*, Macmillan College Publishing Co., New York, 1994.
[7] J. Hertz, A. Krogh, R. G. Palmer, *Introduction to the theory of neural computation*, Addison-Wesley, 1991.
[8] Y. H. Pao, *Adaptive Pattern Recognition and Neural Networks*, Addison-Wesley New York, 1989.
[9] G. L. Bilbro, M. White and W. Synder, "Image segmentation with neurocomputers," in *Neural Computers* eds. R. Eckmiller and C. V. D. Malsburg, Springer-Verlag New York, 1988.
[10] R. P. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Magazine*, pp. 3-22, 1987.
[11] T. Kim, V. Devarajan and M. Manry, "Road extraction from aerial images using neural networks," *Proceedings of ASPRS Annual Convention*, vol. 3, pp. 146-154.
[12] M. Antonucci, B. Tirozzi, N. D. Yarunin et al., "Numerical simulation of neural networks with translation and rotation invariant pattern recognition," *International Journal of Modern Physics B*, vol. 8, no. 11-12, pp. 1529-1541, 1994.
[13] G. A. Carpenter, and W. D. Ross, "ART-EMAP: A neural network architecture for object recognition by evidence accumulation," *IEEE Transactions on Neural Networks*, vol. 6, no. 4, pp. 805-818, 1995.
[14] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological Cybernetics*, vol. 43, pp. 59-69, 1982.

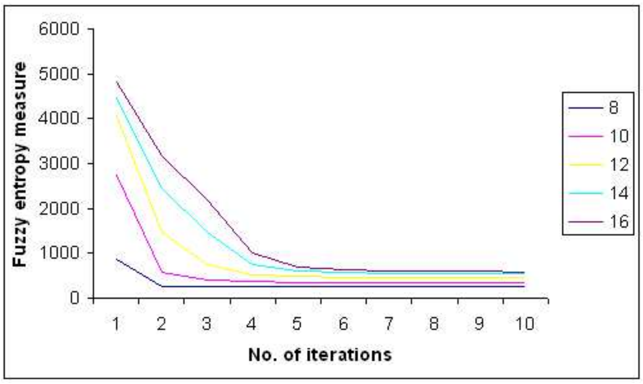

Fig. 25. Variation of fuzzy entropy measures of neighborhood fuzzy hostility indices of synthetic image with time at α=0.5

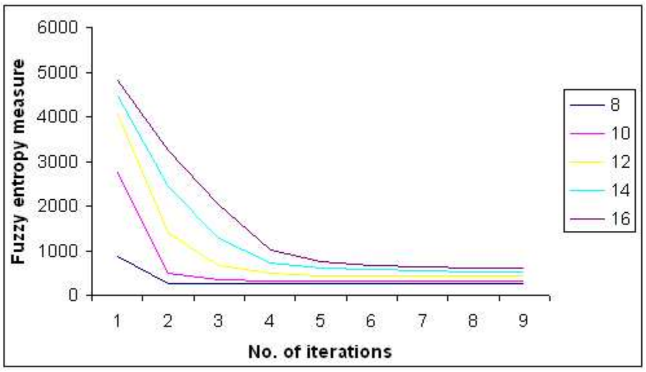

Fig. 26. Variation of fuzzy entropy measures of neighborhood fuzzy hostility indices of synthetic image with time at α=0.75

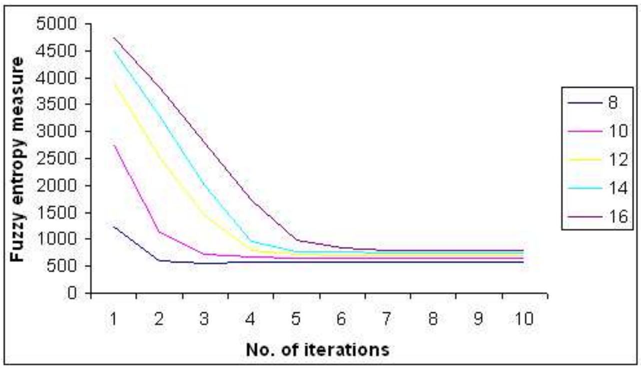

Fig. 27. Variation of fuzzy entropy measures of neighborhood fuzzy hostility indices of spanner image with time at α=0.25

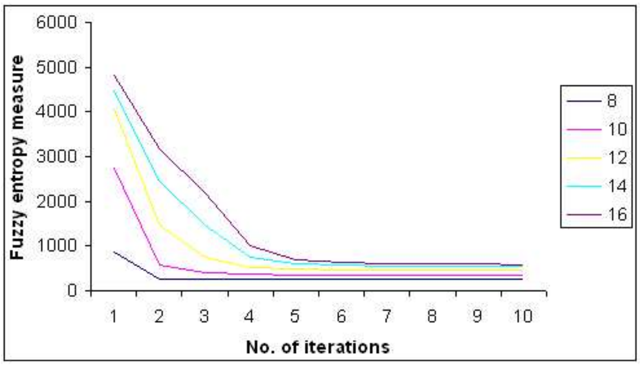

Fig. 28. Variation of fuzzy entropy measures of neighborhood fuzzy hostility indices of spanner image with time at α=0.5

[15] T. Kohonen, "Self-organization and associative memory," Springer-Verlag, London, 1984.

[16] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two state neurons," *Proceedings of Nat. Acad. Sci. U. S.* pp. 3088-3092, 1984.

[17] [6] B. Kosko, "Bidirectional associative memories," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 18, no. 1, pp. 49-60, 1988.

[18] B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*, Prentice-Hall Englewood Cliffs NJ, 1992.

[19] W. Chua and l. Yang, "Cellular network: theory," *IEEE Transactions on Circuits and Systems*, vol. 35, no. 10, pp. 1257-1282, 1988.

[20] W. Chua and L. Yang, "Cellular network: applications," *IEEE Transactions on Circuits and Systems*, vol. 35, no. 10, pp. 1273-1290, 1988.

[21] A. K. Datta, S. Munshi and S. Bhattacharyya, "Object Extraction In Artificial Retina Using Cellular Neural Network Optimized By Genetic Algorithm With Fuzziness Measure," *Proceedings of International Conference on Fiber Optics and Photonics*, vol. 2, pp. 723-725, 2000.

[22] G. A. Carpenter and S. Grossberg, *Pattern Recognition by Self-Organizing Neural Networks*, Cambridge, MA: MIT Press, 1991.

[23] A. Ghosh, N. R. Pal, and S. K. Pal, "Self-organization for object extraction using multilayer neural network and fuzziness measures," *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 1, pp. 54-68, 1993.

[24] R. o. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, New York: Wiley, 1973.

[25] J. T. Tou and R. C. Gonzalez, *Pattern Recognition Principles*, Reading, MA: Addison Wesley, 1974.

[26] S. Bhattacharyya, P. Dutta and U. Maulik, "Multiple Target Tracking Using Self-Organizing Neural Network Based Segmentation of Optical Flow Field," *Proceedings of Conference of Recent Trends In Manufacturing*, pp. 369-376, 2003.

[27] A. Ghosh and A. Sen, "Self organizing neural network for multi-level image segmentation," *Soft Computing Applications to Pattern Recognition and Image Processing*, eds. A. Ghosh and S. K. Pal, pp. 129-144, World Scientific, 2002.

[28] S. Bhattacharyya, U. Maulik and S. Bandyopadhyay, "A Fuzzy Cardinality Based Approximation for Extracting Multi-Scale Objects From Noisy Background Using Self Organizing Neural Network," *Proceedings of International Conference on Communications, Devices and Intelligent Systems*, pp. 461-464, 2004.

[29] S. Bhattacharyya, P. Dutta and U. Maulik, "Multi-Scale Object Extraction Using Self Organizing Neural Network With A Multi-Level Sigmoidal Activation Function," *Proceedings of 5th International Conference on Advances in Pattern Recognition*, pp. 435-438, 2003.

[30] P. Dutta, S. Bhattacharyya and K. Dasgupta, "Multi-Scale Object Extraction Using A Self Organizing Neural Network With A Multi-Level Beta Activation Function," *Proceedings of International Conference on Intelligent Sensing and Information Processing*, pp. 139-142, 2004.

[31] S. Bhattacharyya, P. Dutta and D. DuttaMajumder, "Multiscale Object Extraction Using A Self-Organizing Neural Network With Multilevel Beta Activation Function and its Sigmoidal Counterpart: A Comparative Study," *Proceedings of International Conference on Recent Trends and New Directions of Research in Cybernetics & Systems Theory*, 2004.

[32] S. Bhattacharyya and P. Dutta, "Multiscale Object Extraction with MUSIG and MUBET with CONSENT: A Comparative Study," *Proceedings of KBCS 2004*, pp. 100-109, India.

[33] S. Bhattacharyya and P. Dutta, "XMUBET with CONSENT: A Pixel Hostility Induced Multiscale Object Extractor," *Proceedings of IEEE International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 04)*, pp. 277-282, 2004.

[34] S. Bhattacharyya and P. Dutta, "XMUSIG with CONSENT: Pixel Hostility Induced Multiscale Object Extractor," *Proceedings of International Conference on Intelligent Sensing and Information Processing (ICISIP 2005)*, 2005.

[35] S. Bhattacharyya and K. Dasgupta, "Color Object Extraction From A Noisy Background Using Parallel Multi-layer Self-Organizing Neural Networks," *Proceedings of CSI-YITPA(E) 2003*, pp. 32-36, 2003.

[36] S. Bhattacharyya and P. Dutta, "A Parallel Self-Organizing Neural Net-

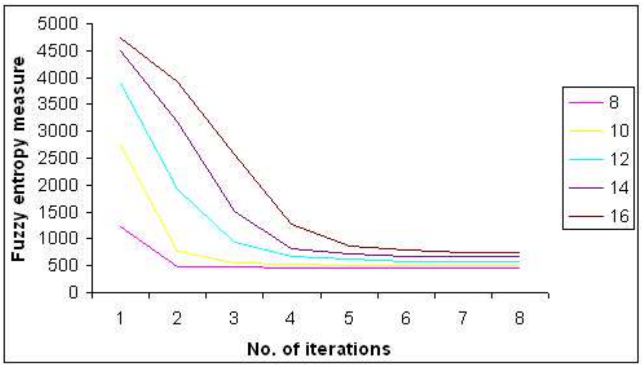

Fig. 29. Variation of fuzzy entropy measures of neighborhood fuzzy hostility indices of spanner image with time at α=0.75

work Architecture For Extraction of Graded Color Objects," *Proceedings of International Conference on Recent Trends and New Directions of Research in Cybernetics & Systems Theory*, 2004.

[37] S. Bhattacharyya, P. Dutta and U. Maulik, "Graded color object extraction by a Parallel Self-Organizing Neural Network (PSONN) architecture guided by a MUBET activation function," *Proceedings of ICIS 2005*, 2005.

[38] S. Bhattacharyya, P. Dutta and U. Maulik, "MUSIG and MUBET guided PSONN based color object extraction: A Comparative Study," *Proceedings of International Conference on Information Technology 2005*, 2005.

[39] S. Bhattacharyya, P. Dutta and P. Nandi, "True Color Object Extraction By A Parallel Self Organizing Neural Network (PSONN) Architecture Guided By XMUBET With CONSENT," *Proceedings of EAIT 2006*, 2006.

[40] S. Bhattacharyya and P. Dutta, "Designing pruned multilayer self organizing neural network (MLSONN) for object extraction from a noisy background," *Book of Abstracts of SOOP 05*, pp. 131-132, 2005.

[41] S. Bhattacharyya and P.Dutta, "Designing pruned neighborhood neural networks for object extraction from noisy background," *International Journal of Foundations of Computing and Decision Sciences*, vol. 31, no. 2, pp. 105-134, 2006.

[42] L. A. Zadeh, "Fuzzy sets," *Inform. and Control*, vol. 8, no. 1, pp.338-353, 1965.

[43] T. J. Ross and T. Ross, *Fuzzy Logic With Engineering Applications*, McGraw Hill College Div., 1995.

[44] A. Deluca and S. Termini, "A definition of non probabilistic entropy in the setting of fuzzy set theory," *Information and Control*, vol. 20, pp. 301-312, 1972.

Siddhartha Bhattacharyya did his Bachelors in Physics and Optics & Optoelectronics from Calcutta University, Kolkata, India in 1995 and 1998 respectively. Subsequently, he did his Masters in Optics & Optoelectronics in 2000 from Calcutta University, Kolkata, India.
He is currently a Lecturer in the Department of Computer Science and Information Technology of University Institute of Technology, The University of Burdwan, Burdwan, India. He was a Lecturer in the Department of Information Technology of Kalyani Government Engineering College, Kalyani, India during 2001-2005. He has served as a Project Fellow in Bengal Engineering and Science University, Shibpore, India. He is a co-author of a book and about 30 research publications. His research interests include soft computing, pattern recognition and image processing.
Mr. Bhattacharyya is a Fellow of **OSI**, India.

Paramartha Dutta did his Bachelors and Masters in Statistics from Indian Statistical Institute, Kolkata, India in 1988 and 1990 respectively. Subsequently, he did his Masters in Computer Science in 1993 from Indian Statistical Institute, Kolkata, India. He did his Ph.D in 2005 from Bengal Engineering and Science University, Shibpore, India.
He is currently an Assistant Professor in the Department of Computer Science and Engineering of Kalyani Government Engineering College, Kalyani, India. He was an Assistant Professor and Head of the Department of Computer Science and Engineering of College of Engineering and Management, Kolaghat, India during 1998-2001. He has served as a Research Scholar in the Indian Statistical Institute, Kolkata, India and in Bengal Engineering and Science University, Shibpore, India. He is a co-author of 4 books and about 50 research publications. His research interests include evolutionary computing, soft computing, pattern recognition, multiobjective optimization and mobile computing.
Dr. Dutta is a Fellow of **OSI**, India. He is the member of **ISCA**, **CSI** and **IETE**, India.

Ujjwal Maulik did his Bachelors in Physics and Computer Science in 1986 and 1989 respectively. Subsequently, he did his Masters and Ph.D in Computer Science in 1991 and 1997 respectively. He is a senior member of **IEEE**, USA.
He is currently a Professor in the Department of Computer Science and Engineering of Jadavpur University, Kolkata, India. He was the Head of the Department of Computer Science and Engineering of Kalyani Government Engineering College, Kalyani, India during 1996-1999. He has worked in the Center for Adaptive Systems Application, Los Alamos, New Mexico, USA in 1997, University of New South Wales, Sydney, Australia in 1999, University of Texas at Arlington, USA in 2001, University of Maryland Baltimore County, USA in 2004 and Fraunhofer Institute AiS, St. Augustin, Germany in 2005. He has also visited many Institutes/Universities around the world for invited lectures and collaborative research. He is a co-author of 2 books and about 100 research publications. He has been the Program Chair, Tutorial Chair and a Member of the program committee of many international conferences and workshops. His research interests include artificial intelligence and combinatorial optimization, soft computing, pattern recognition, data mining, bioinformatics, VLSI and distributed systems.
Dr. Maulik is a Fellow of **IETE**, India. He is the recipient of the Govt. of India **BOYSCAST** fellowship in 2001.

Prashanta Kumar Nandi did his Bachelors in Electrical Engineering from Jadavpur University, Kolkata, India in 1965. Subsequently, he did his Masters in Electrical Engineering from Calcutta University, Kolkata, India in 1967. He did his Ph. D. in Electrical Engineering from Indian Institute of Technology, Kanpur, India in 1982.
He is currently the Program Director of NRC in the Department of Computer Science and Technology of Bengal Engineering and Science University, Shibpore, India. He was a Lecturer and Assistant Professor in the Department of Electrical Engineering of Bengal Engineering and Science University, Shibpore, India during 1969-1978 and 1978-1991 respectively. He was a Professor in the Department of Computer Science and Technology of the same University during 1992-2005. He held the post of Adjunct Professor in the Department of Information Technology of Bengal Engineering and Science University, Shibpore, India during 2005-2006. He has several research publications to his credit.