Hopfield Neural Network for Change Detection in Multitemporal Images

Sneha Bishnoi
V.E.S. Institute of Technology
University of Mumbai
Mumbai-74

Vijay Gaikwad
V.E.S. Institute of Technology
University of Mumbai
Mumbai-74

Saurabh Asegaonkar
V.E.S. Institute of Technology
University of Mumbai
Mumbai-74

ABSTRACT
This paper proposes a supervised change detection technique for multitemporal remote sensing images. The technique is presented after studying three different models for change detection using neural network and assimilating the unique feature of each of the model. The technique is based on Hopfield neural network modified to model spatial correlation between neighboring pixels of the difference image. Each pixel in the difference image is represented by a neuron in the Hopfield network that is connected to its neighbors. These connections to the neighboring units model the spatial correlation between pixels and are assigned weights according to their influence on each other with help of training sets. The information about the status of the network is rendered through an energy function allocated to the network. A threshold is defined for segmenting the pixels into two classes of pixel-changed and unchanged. Change detection map is obtained by iteratively updating the output status of the neurons until a minimum of the energy function is reached and the network assumes a stable state. Experimental results carried out on two multispectral multitemporal remote sensing images confirm the effectiveness of the proposed technique.

General Terms
Image processing, Neural Networks, Pattern Detection

Keywords
Change detection, Hopfield neural network, Thresholding, Remote sensing, Image differencing.

1. INTRODUCTION
Change detection in images obtained from remote sensing is mainly used to identify changes that have occurred over a period of time in a specific area. This is done by comparing the images of a specific geographical area, obtained at same angle at different times and identifying the difference in the images to detect change. Detection of land-cover changes is one of the most interesting aspects of the analysis of multitemporal remote sensing images. In particular, it is very useful in many applications, like land use change analysis [3], study on shifting cultivation, monitoring of pollution, [4] urban growth, assessment of burned areas, assessment of deforestation or other structural damage due to disaster [2], and so on. Many of these applications relate to the analysis of large areas on the Earth surface; then, it is important to exploit automatic techniques to detect land-cover changes in order to reduce the effort required by manual image analysis.

The change detection techniques include three main steps before actual detection 1) Pre-processing–this includes processes like coregistration [5], noise reduction and radiometric corrections. These processes are required to make the two images as comparable as possible. 2) Image differencing- The next step is the finding out the difference between the two coregistered images. For finding out the difference between two images we can either adopt a simple technique for single spectral images by differencing the values of corresponding pixel value. For multi spectral images vector the difference operator applied is a vector and hence CVA technique [6] is used. Once the images have been subtracted and difference obtained, next step would be 3) Analysis of Image-The change detection process occurs in this step. The difference obtained is compared with a minimum threshold to identify whether the change is significant. To understand in simple terms let us consider $I_1(x,y)$ and $I_2(x,y)$ to be the intensities of the pixel at coordinate (x,y) in image 1 and 2 respectively. The difference would thus be $D(x,y)= I_1(x,y) - I_2(x,y)$. The final changed image would be F(x,y) where,

$$F(x,y) = \begin{cases} 1, & |D(x,y)| >= T \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

T is the threshold which can be obtained by manual trial and error method or with automated techniques.

Comparison of algorithms for change detection has shown that neural networks produced acceptably high accuracy for land cover change detection for training and test data sets [7]. The slight inefficiency in final area wide results while using neural networks can be blamed to the conversion of the trained network to mapping information rather than to the method itself. When developing algorithms for change detection using neural networks almost all models utilize the Hopfield neural network architecture. In the studied models each neuron corresponds to a pixel of the difference image and is connected to all the neurons in the neighborhood. Each neuron is given an initial state (bias) which is modified iteratively until the network reaches a stable state (minimum of the energy function calculated after each iteration). During the process the difference image is classified into two groups-neurons with value 1 indicating changed pixel and neurons with value -1 indicating unchanged pixels.

Major advantages of such techniques are: 1) It exploits the spatial contextual information making it robust to unrelated noise patterns. 2) It does not require manual setting of any parameter. 3) Initialization of threshold and subsequent change of energy function value are independent of any specific parameter. 4) Best performance in terms of accuracy is obtained with neural network models.

2. HOPFIELD NEURAL NETWORK

The network consists of a set of neurons each connected to its neighbours. Each neuron is provided with an initial state. The output of each neuron is provided to its neighbours according to the synaptic weight of connections to its neighbours.

$$a_i \longleftarrow \begin{cases} 1 & \text{if } \sum_j w_{ij} s_j > \theta_i, \\ -1 & \text{otherwise.} \end{cases} \qquad (2)$$

$$a_i \longleftarrow \begin{cases} 1 & \text{if } \sum_j w_{ij} s_j > \theta_i, \\ 0 & \text{otherwise.} \end{cases} \qquad (3)$$

Where:

- w_{ij} is the strength of the connection weight from unit j to unit i (the weight of the connection).
- s_j is the state of unit j.
- θ_i is the threshold of unit i.

The connections in a Hopfield net typically have the following restrictions:

- $w_{ii} = 0$ (no unit has a connection with itself)
- $w_{ij} = w_{ji}$ (connections are symmetric)

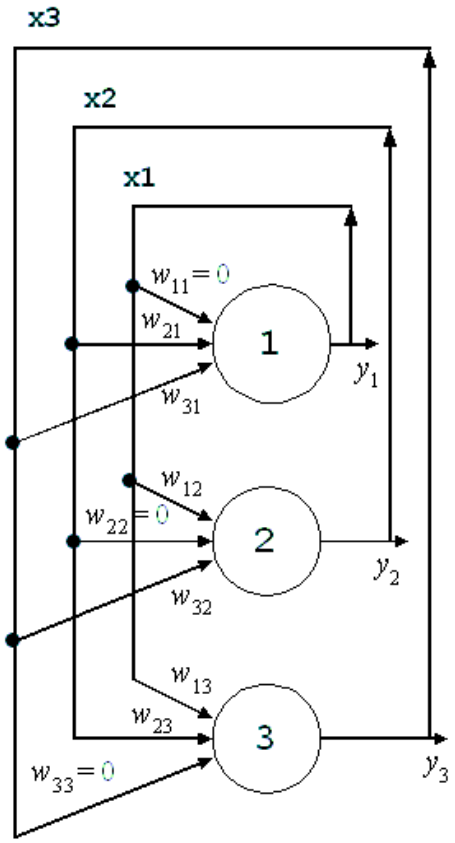


Fig. 1: A Hopfield structure showing three neurons with connections with each other

The requirement that weights should be symmetric is provided so that the energy function decreases monotonically. Hopfield network have a scalar value associated with each state of the network referred to as the "energy", E, of the network, where:

$$E = -\frac{1}{2}\sum_{i,j} w_{ij} s_i s_j + \sum_i \theta_i s_i \qquad (4)$$

The value is so chosen because it ensures that if units are randomly chosen to update their activations, the network will ultimately converge to configuration of states which give a local minimum value to the energy indicating a stable state

Hopfield has two types of models-

2.1 Continuous model

In case of continuous model, the output value of each neuron can take values between -1 to +1. In this model the activation function must satisfy the certain conditions, it should be monotonic non-decreasing function and its inverse should exist. A typical choice of the function $g(U_i)$ i.e. activation function is

$$g(U_i) = \frac{2}{1 + e - (U_i - \tau)} - 1 \qquad (5)$$

2.2 Discrete model

In this model neurons have output as either +1 or -1 and nothing else. In this case activation function can be defined as having value +1 if the output of the neuron is above threshold or else -1.

$$Vi = g(Ui) = \begin{cases} +1, \text{ if } Ui \geq \theta i \\ -1, \text{ if } Ui < \theta i \end{cases} \qquad (6)$$

where, θi is the predefined threshold value of neuron i.

3. OVERVIEW OF THE MODELS

Here we will analyze three models of Hopfield neural network for change detection in order to understand the advantages of each and how they can be combined to form a more accurate model.

3.1 Hopfield network with self-loop

This approach is based on the continuous Hopfield neural network (HNN) model for solving the image change detection problem between two images. A difference image is obtained by subtracting pixel by pixel both images. The network architecture is built so that each pixel in the difference image is a node in the network. Each node is characterized by its state (between +1 and -1), which determines if a pixel has changed. The energy function is calculated. This function ascertains that the network converges to a configuration of states that are stable. The HNN model allows each node to take on continuous state values, which determine the strength of the change.

In most of the neural network approaches, the value of the energy function is calculated taking into account only inter-pixel relations between a pixel and its neighbors. This implies that a pixel is labeled as changed or unchanged according to the information supplied by its neighbors and that its own information is ignored. This HNN model overcomes this drawback and for each pixel allows one to achieve a tradeoff between the influence of its neighbors and its own value. This effect is recorded in the energy function to be minimized.

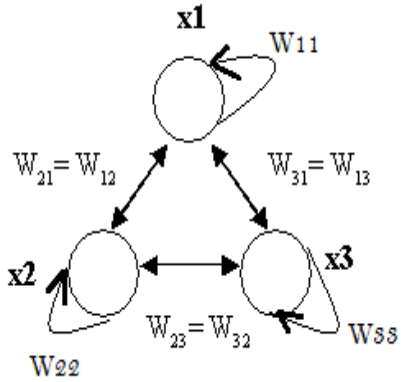


Fig. 2: Hopfield Network with self loop

The proposed method has proven to be robust against noise as compared with the other methods. Its accuracy and performance against the models without self loop was found to be higher.

3.2 Unsupervised Context-sensitive technique based on modified Hopfield

Consider two coregistered multispectral images I_1 and I_2 of size p × q, acquired over the same area at different times T_1 and T_2, and let

$$D = \{l_{mn}, 1 \le m \le p, 1 \le n \le q\}$$

be the difference image obtained by applying the CVA technique to I_1 and I_2. We assign to each spatial position or in other words the pixel with position (m,n)∈D a neuron of the network. The spatial relation between neighboring pixels is modeled by defining the neighborhood systems N of order d, for a given spatial position (m, n) as $N_{dmn}=\{(m, n)+ (u, v); (u, v)\in N_d\}$. The neuron in position (m, n) is connected to all its neighboring units included in N_d. According to the value of d, the neighborhood system assumes different configurations. Here, only the first (four neighboring neurons) and second-order (eight neighboring neurons) neighborhood systems have been considered.

Let $W_{mn,uv}$ be the weight denoting the connection strength between the $(m, n)^{th}$ and $(u, v)^{th}$ neurons. We assume that $W_{mn,uv}$ = 1 if (u, v) $\in N_d$ otherwise, $W_{mn,uv}$ = 0. Hence, the presented topology can be seen as a modified version of the Hopfield network in which the connection strength (weight) to all neurons outside the neighborhood (N_d) is zero. As each neuron is connected only to its neighbor, the output of a neuron depends only on its neighboring elements. In this way, the network is able to model and consider the neighboring information of each pixel.

The energy function defined for the architecture consists of 3 parts- The first part is the feedback, whereas the second and third parts correspond to the input bias I_i and the threshold value θ_i of each neuron in the network, respectively. Thus, taking into account of all pixels, the total energy can be written as-

$$E=-\sum_{m=1}^{p}\sum_{n=1}^{q}\sum_{(u,v)} W_{mn,uv}V_{mn}V_{uv}-\sum_{m=1}^{p}\sum_{n=1}^{q} I_{mn}V_{mn} \quad (7)$$

In the case of the discrete model, both the initial external input bias I_{mn} and the input U_{mn} to a neuron are taken as +1if the gray value of the corresponding pixel of D is greater than initial threshold value t; otherwise, they have a value of −1.In the continuous model, both the initial external input bias and the input to a neuron at position (m, n)are proportional to $(l_{mn}/t) - 1$ (if $(l_{mn}/t) - 1 > 1$ then the value +1 is used for initializing the corresponding neuron.

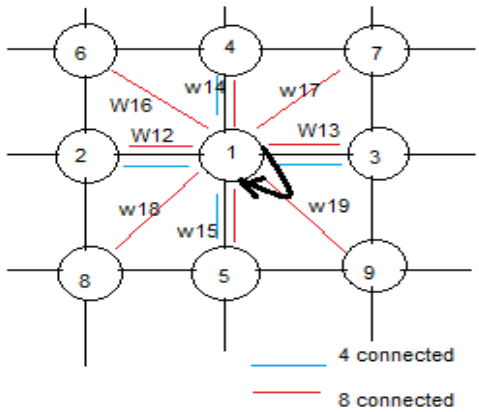


Fig. 3: Hopfield Network with context based connections

3.3 Hopfield Network with unsymmetric connections

Classical approaches in the Hopfield networks have been designed assuming that the weights between two nodes are symmetric. The main purpose of this technique is to see the result when we assume that the weights need not be symmetric. This model is based on the assumption that different nodes could have different levels of relevance. So, assuming that the node *i* is more relevant than the node *j*, the influence of *i* over *j* should be greater than the influence of *j* over *i*. The relevance is introduced based on the assumption that the value of a relevant node should remain unchanged during the iterative process.

Such networks consist of nodes and the various causal relations (arcs) that exist between these nodes. Each relation is accompanied by a weight that defines the type of binary relation between the two nodes. The causal weights *Wij* are assigned values in the interval [-1, +1]; *Wij* = 0 indicates no causality. Positive relation (*Wij* > 0) between two nodes *Mi* and *Mj* indicates: *Mi* increases as *Mj* increases and *Mi* decreases as *Mj* decreases. Negative causal relation (*Wij* < 0) indicates: *Mi* decreases as *Mj* increases and *Mi* increases as *Mj* decreases. These networks however, have the other restriction and thus no self loop or no feedback from a node to itself is allowed, so *Wii* = 0.

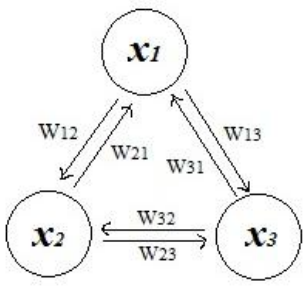


Fig. 4: Hopfield Network with unsymmetrical connections

4. CHANGE DETECTION

The image change detection problem is solved as follows: given two images $I_1(x,y)$, $I_2(x,y)$ of size $M \times N$ of the same area under consideration, taken at different times; the aim is to detect if a pixel, located at (x,y), has changed or not and if yes, then the magnitude of the change. We build a network of $n = M \times N$ nodes, where each node i represents a pixel location (x,y). We create the n x n weight matrix W, assuming that it could be unsymmetric. At each iteration k, the node i in the network has the activation level Ai^k ranging in [-1, +1]. This implies that every node has a positive or negative valued state that determines the magnitude of the change at each pixel location.

5. PROPOSED TECHNIQUE

The technique proposed overcomes the drawbacks of above models. A modified discrete Hopfield model with self-feedback which ensures that node's own information is considered in its classification. The neighborhood of a node is formed by 12 neighboring pixels thus utilizing third order connectivity. The connection to each of the neighboring nodes is assigned an initial weight between 0 and 1.

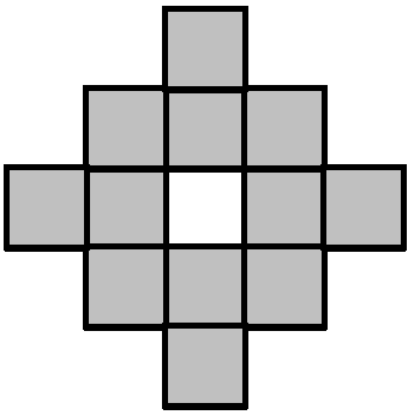


Fig. 5: Third order connectivity with 12 neighboring pixels

The initial weights to the connection are assigned based on the distance of the 12 neighbors to the node. This would be altered later by help of training classes and data sets. In order to compare two images of same geographical area at two different time, we would first collect images of same location in intermediate time period and apply the technique iteratively on those images. The processing of each intermediate image will train the algorithm to map the influence of two neighboring pixels in form of asymmetric weights. For example, if pixel i and j are neighbors with initial weight of their connection as Wij=Wji. At the end of intermediate processing suppose we obtain Wij>Wji , this implies that pixel i has a greater influence on changing pixel j than vice versa.

Thus unsymmetrical weights to two neighboring nodes would be assigned depending on past history. This implies that the training algorithm would assign the weights to twelve neighboring connections irrespective of their distance. Thus the false classification of an unchanged pixel into changed group due to effect of its neighboring pixels would be eliminated.

6. IMPLEMENTATION AND RESULTS

6.1 Description of Experiments

A simple model based on modified Hopfield technique on multitemporal remote sensing images has been implemented. The implementation involves use of .NET4.0 framework technology and the graphics libraries of C# programming language to implement the algorithms. In the experiment, two different data sets related to areas in Mexico are used. Change detection map was generated using discrete Hopfield network considering second order neighborhood.

The experiment consisted of three stages, (1) Image differencing (2) Thresholding (3) Application of Hopfield network algorithm. Initially two images of the same size, of the same area captured over different time instances were taken. The two images were converted to greyscale and other pre-processing transformations were applied to it, like brightening, contrasting and filtering. After the two images were comparable to each other, the image difference was found out by subtracting the pixel intensity of one image from other image. This gives us the difference image D which corresponds to the pixel change in the two considered images. To this difference image a Thresholding operation is performed which segments the difference image into changed and unchanged regions. The threshold value for pixel value difference was set to 40 by a manual trial-and-error procedure (according to a tradeoff between false and missed alarm). Since we are using a discrete model, the value of the pixel in difference image is set to 255 (since we are using grayscale images) corresponding to Boolean 1 logic, if the difference value is greater than threshold or it is set to 0 otherwise corresponding to Boolean logic 0. This transforms the difference image into a bipolar image with pixel having value 255 (white) if they are changed pixels or 0 (black) if they are unchanged pixels.

Once the segmented image is obtained, second order Hopfield network is iteratively applied over the image till energy function becomes stable. In our implementation model we assigned weights to the connections of the neighboring pixels by default as 1. The energy function for each pixel was given as:

$$E = \sum_{m=1}^{p} \sum_{n=1}^{q} W_{mv} V_{mn} \qquad (8)$$

where, W_{mn} = 1 for second order neighborhood

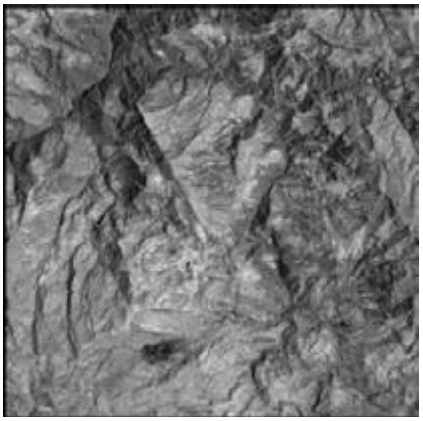
$$V_{mn} = \begin{cases} +1, \text{ if pixel value at (m,n) is 255} \\ -1, \text{ if pixel value at (m,n) is 0} \end{cases}$$

i.e. if the energy value of a pixel is positive (>=0), which implies that most of its neighbors are changed pixels, then the pixel is set as changed pixel and if the energy value of a pixel is negative (<0), which implies that most of its neighbors are unchanged pixels, then the pixel is set as unchanged pixel. This process is iteratively applied to each pixel till none the pixels change their state and the energy function reaches its minimum.

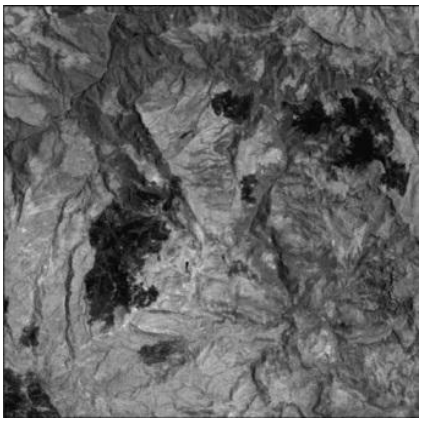
6.2 Results and Outputs

The output images at each stage of processing were obtained and compared. The final change map was obtained which gives us the desired changes between the two input images. This final change map was easier to interpret and analyze, than the mere difference image because it cropped the insignificant minute changes.

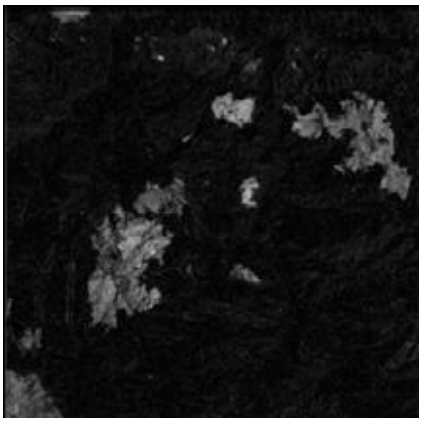
Further models can be also developed as per our proposed technique which makes use of asymmetric weights and third order neighborhood which could make the system more accurate and robust to noise and unwanted changes.



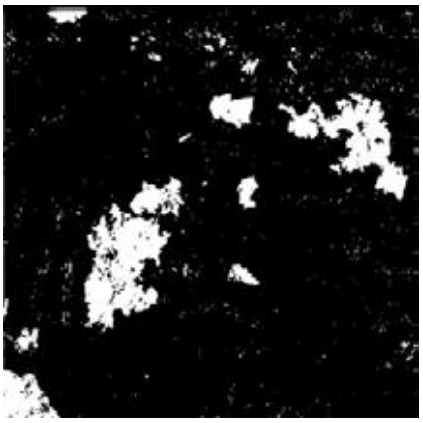
(A)



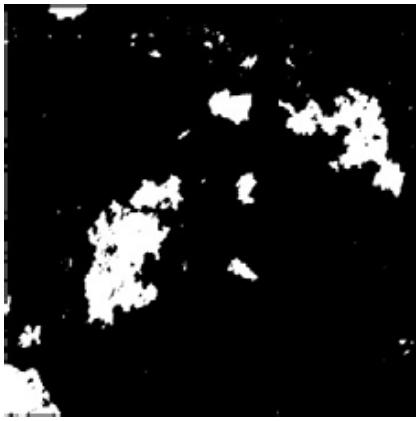
(B)



(C)



(D)



(E)

Fig. 6: Images of Mexico. (a) Band 4 of the Landsat ETM+ image acquired in April 2000, (b) band 4 of the Landsat ETM+ image acquired in May 2002, (c) corresponding difference image generated by image differencing technique, (d) segmented image generated by Thresholding of difference image, and (e) final change map obtained after applying discrete second order Hopfield networks over threshold image

7. DISCUSSIONS AND CONCLUSIONS

In this paper, an unsupervised context sensitive technique for change detection in multitemporal images has been proposed. The technique models the spatial correlation between neighboring pixels in the difference image by using a Hopfield neural network implemented according to a specific architecture, where connections between neurons are properly defined. The architecture of the network represents the difference image structure by associating a neuron to each pixel; this allows to easily define the spatial neighborhood of each neuron and, accordingly, to properly define the weights of the connections among different units (pixels). On the basis of this architecture, an energy function associated with the state of the network is defined that models the information present in both the radiance of the difference image pixels and the spatial context of each unit. The state of the network is initialized according to a simple thresholding of the difference image; the threshold value is derived according to a heuristic yet effective threshold selection procedure.The presented technique shows the following advantages with respect to the reference context-sensitive method: 1) It is distribution free, i.e. it does not require any explicit assumption on the statistical model 2) it does not require the setting of any input parameter 3) it is more robust to minor changes and errors in the images, and thus intelligently sorts desired changes from unwanted changes.Experimental results obtained on different real multitemporal data sets confirm the effectiveness of the proposed approach, which significantly outperforms the standard optimal-manual context-insensitive technique.

8. ACKNOWLEDGMENTS

We would like to thank our B.E project guide and the co-author Mrs. Sharmila Sengupta (M.E EXTC) Asst. Professor, Computer Engineering Department(sharmilase@yahoo.com), VESIT and our teachers; Dr. Nupur Giri (M.E.Computers, Phd.), HOD Computer Engineering Department, VESIT and Mrs. Saylee Gharge (M.E, Phd- Image Segmentation from NMIMS Mumbai), Lecturer, Department of Electronics and Telecommunications, VESIT for their constant guidance, support, motivation and insightful comments that helped to improve the quality of this work..

9. REFERENCES

[1] A. Singh, “Digital change detection techniques using remotely sensed data,” Int. J. Remote Sens., vol. 10, no. 6, pp. 989–1003, 1989.

[2] J. A. Richards and X. Jia, Remote SensingDig ital Image Analysis, 4th ed.Berlin, Germany: Springer-Verlag, 2006.

[3] J. Cihlar, T. J. Pultz, and A. L. Gray, “Change detection with syntheticaperture radar,” Int. J. Remote Sens., vol. 13, no. 3, pp. 401–414, 1992.

[4] L. Bruzzone and S. B. Serpico, “An iterative technique for the detection of land-cover transitions in multitemporal remote-sensing images,” IEEETrans. Geosci. Remote Sens., vol. 35, no. 4, pp. 858–867, Jul. 1997.

[5] Francesco Serafin-“SAR Image Coregistration Based onIsolated Point Scatterer” IEEE Geosci. ,Remote Sens., vol. 3, no. 3,, Jul 2006.

[6] P.G. Silva, J.R. Santos, Y.E. Shimabukuro, P.E.U. Souza-“Change Vector Analysis technique to monitor selective logging activities in Amazon” 0 7803-7929-2/03/$17.00 (C) 2003 IEEE

[7] D. L. Civco, J. D. Hurd, E. H. Wilson, M. Song, Z. Zhang-“A comparison of land use and land cover change detection methods” Proceedings 2002 ASPRS Annual Convention (22 April 2002)

[8] The IEEE website. [Online]. Available: http://www.ieee.org/

[9] Wikipedia:www.wikipedia.org/