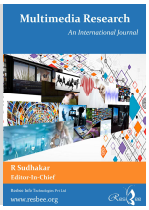
Vol.6 No.1 Jan. 2023
Multimedia Research
Received 29 September, Revised 12 November, Accepted 28 December



Forest Change Detection and Prediction using the Crow Chicken Swarm Optimization based Deep LSTM Classifier

M Naveen Kumar
Department of CSE
Telangana University, Nizamabad, Telangana, India.

Abstract: The detection of the changes in the earth surface is an important factor, as it enables the understanding regarding the interaction and relationship between the natural and human phenomena for achieving better decision-making strategy. Due to the socio-economic factor, there exist frequent changes in the forest area. Various methods are introduced to predict the change detection in the forest area, but to accurately find the change prediction in the forest region is a challenging task in the research community. Therefore, an effective method named Crow-Chicken Swarm Optimization algorithm based Deep Long Short term Memory (C-CSO based Deep LSTM) is proposed to find the forest change detection and prediction. The proposed algorithm uses the intelligent behaviour of crows with the hierarchical order and the mimicking behaviour of chicken groups to enhance the effectiveness of prediction in forest area. The time-series data is generated from the result of change prediction in order to increase the performance of the Deep LSTM classifier. The proposed C-CSO based Deep LSTM attained better performance in terms of the metrics, like Mean Square Error (MSE), and accuracy with the values of 0.0014 and 81.200%, respectively.

Keywords: Active Contour Approach, Change Detection, Change Prediction, Fuzzy Non local Means (Fuzzy NLM) filter, Vegetation Indices.

1. Introduction

The socio-economic and the ecosystem management studies at the national, international, and regional scale involve the process to monitor and detect the forest changes [1]. Forests are the livelihood ecosystems that are categorized by the anthropogenic and natural processes, and thus these process are constantly changes. Therefore, it is significant to introduce the methods to monitor the forest areas for revising the inventories of forest to analyze, plan, and to manage the health of forest areas to detect the growth rate. The remote sensing data obtained from the forest areas at various time provides suitable data source to achieve the automatic detection changes in forest [8]. In general, detecting the forest changes is a complex factor to continuously monitoring the environmental changes and to investigate the environmental issues, like biodiversity loss, deforestation, and depletion of natural resources [1]. Deforestation is the process of converting the forest land to the non forest area. However, ranching, road building, fire, logging, and converting the forest to farming area are some of the effective actions that devastate the influence of forests in worldwide. The consequences and the impact of deforestation extend the boundaries of forest land. Some of the deforestation effects are increasing climate change in the rate of soil erosion, global warming, extinction of species, and absorption of greenhouse gases [10]. However, the changes in the forest area affect the agricultural production, water resources, and resource management system [9]. The deforestation not only minimizes the ecological integrity and biomass stock but also aggravates the food damage. The monitoring and the quantification process over time at both inside and surrounding the target areas are crucial for the conservation efforts [2].

Accurate and timely made changes in the detection of earth surface are an important factor to understand the interaction and relationship between the natural and human phenomena to achieve enhanced decision making. The remote sensing data is the primary source used to detect the changes in the recent years. The change detection is defined as the process of observing the states of object or identifying the differences at varying time. The change detection process contains the multi temporal dataset for analyzing the temporal effect of phenomenon. The benefits of data acquisition, digital format, and its synoptic sight make the phenomenon appropriate to computer processing [11]. The digital change detection contains the temporal quantification phenomena from the multi date imagery, which is

https://doi.org/10.46253/j.mr.v6i1.a2

retrieved from the satellite based sensors. The literature works shows that the change detection in the digital form is a complex task to achieve. The interpreter that is used to analyze the large scale aerial based photography always generates accurate result with high degree of precision. Moreover, visual change detection is complex to replicate, as various interpreters generate various results [12]. Information gathering and forest monitoring are basically achieved through the time consuming and expensive field studies that combines the aerial and terrestrial survey using optical sensors. With the focus of national and regional scale applications, very high resolution (VHR) stereoscopic acquisition provides faster, more remote sensing capability, and less expensive than the aerial and terrestrial ones. The satellite systems provides more benefit to the aerial acquisition, like higher ground coverage capacity, more spectral bands available, frequent repetition, and no over flight permission required in the area of interest [4].

To develop the change detection techniques to predict the forest changes plays a major challenge in remote sensing environment [13]. Remote sensing based on the satellite image is highly used in detecting the forest changes and to modify the forest maps. Various change detection methods are introduced in the early days of ground observation. They are categorized into three different types, as visual interpretation, pixel-based method, and object-based method. The visual interpretation based on the single or multi date images uses computer assisted expertise for labelling and delimiting the zones. The digital change detection approach offers high repeatable and quantitative information than the visual interpretation. In the recent decades, the object based approach is introduced to perform the change detection by combining the quantitative feature of pixel based method with the contextual testing of visual interpretation [14]. The pixel-based detection method uses change vectors [19], local texture [16], vegetation index [17] [18], and spectral mixture analysis [20]. The machine learning classifiers, such as support vector machine (SVM) [15], decision trees [22], and multilayer perceptron [21], are used in the pixel based detection. These methods use the handcrafted features with the domain expertise. However, the prevalent method used in the change detection with the remote sensed data is classified into two classes, as object-based method, and low level local methods. The low level method uses the statistical indices that are extracted from the spectral image pixel values. These approaches are restricted to perform the pixel level analysis, such that they are agnostic to the contextual information [1].

This research is focus on the forest change detection mechanism using the proposed C-CSO based Deep LSTM classifier with respect to the time-series data. The proposed method identifies the percentage of change detection based on the cosine similarity measure through the satellite images. The proposed classifier considered the segments with high similarity measure and segments collected from the images at different instances are mapped together in order to estimate the forest and non-forest region. The major functionality of Deep LSTM classifier is that it effectively detects the changes in the forest area and increases the prediction performance based on the time-series data. The fitness function is evaluated based on the location of the crow such that the fitness value with minimal error value is accepted as the optimal solution. Moreover, the position update equation of CSA is modified with the standard equation of CSO to obtain the updated equation of the proposed algorithm. Based on the optimal value, the proposed method detects the changes, which in turn reflect to make the best and accurate prediction in the forest region.

The major contribution of this research is explained as follows:

- The main contribution of this research is regarding the forest change detection and prediction using the proposed C-CSO based Deep LSTM classifier, which effectively detects the changes in the forest location based on the satellite images collected from the same location at different time instances.
- The proposed optimization named C-CSO inherit the characteristic features of Chicken swarm optimization (CSO) and Crow search algorithm (CSA), which in turn detect the changes with respect to the pixel based and segment based change detection strategies using time-series data. Moreover, it facilitates the property of biophysical relevant features and achieves better prediction performance with relatively low operational cost.

The rest of the paper is organized as follows: Section 2 describes the existing methods of forest change detection. Section 3 elaborates the proposed algorithm to compute the percentage of change detection in the forest area, and section 4 describes the results and discussion of the change prediction method. Finally, section 5 concludes the paper.

2. Motivation

In this section, various existing change detection methods are surveyed along with their merits and demerits.

2.1 Literature survey

Various existing forest change detection methods are reviewed. Khan, S.H *et al.* [1] introduced a forest change detection method to perform the change analysis with the learning features and temporal resolution from the reflectance data. It uses the multi resolution profiles to identify the candidate set at object level. It achieves better prediction results, but was not suitable with the large scale annotated set. Karimi, N *et al.* [2] developed a spectral mixture analysis (SMA) model to estimate the images with the selected components. It simulates the direction of changes between the temporal images. It uses the vector analysis scheme to detect the forest changes. It was robust with the wide range of conditions, and required more effort to acquire the atmosphere information. Chance, C.M *et al.* [3] introduced a sun canopy sensor correction algorithm to detect the replacing changes. Based on the date of acquisition, the pixel values were modified. This method was robust in the illumination condition and spectral response to detect the change. It failed to explore the similar changes arise in the detection process. Stylianidis, E *et al.* [4] introduced a 3D forest monitoring system named FORSAT to provide the spatial information in the forest area with long term monitoring. It effectively analyzes the satellite images and extracts the quantitative information, such as volume, measurements, and area of deforestation. Here, the change output is generated through the satellite sensor image. It attained reliable solution in the forest operations and forest management. Due to irregular location, the change detection was not effectively achieved. Karila, K *et al.* [5] introduced a TanDEM-X digital surface model was used to achieve the change detection. Here, the density and the height variations in forest were detected using the bi-temporal model. Based on the changes detected in the height, the changes in the ground biomass were estimated. It effectively detects the biomass changes with different magnitude order, but the quality of input data was poor. Vazquez-Quintero, G *et al.* [6] introduced a markov chain model to perform the projection and detection of forest changes. The temporal and the spatial dynamics used the temperate model to detect the changes. The kappa coefficient was used to validate the land classification. It provides more information in the management of forest ecosystems. However, the global climate changes leads to high error. Healey, S.P *et al.* [7] developed a stacked generalization scheme to detect the forest change. It uses the signal processing to extent the detection of surface phenomena. It allows efficient processing with large dataset, but the computation load of this scheme was high. Marinelli, D *et al.* [8] developed an object based detection model to detect the changes in forest areas. It monitors the growth of crown volume and treetop to find the changes in single tree. This method accurately detects the changes, but the false and missed alarm rate was very low.

In 2023, Yin *et al.* [29] have used Siamese Network (SAGNet) to get highly resolute images. Initially, a convolution dual-stream algorithm was used to obtain bit emporal images with high semantic features. Then these features were used to separate semantic variation data from the global semantic aggregation module (GSAM). Moreover, the background noise was completed and removed through the bilateral feature and cross-scale fusion module. This obtained the accuracy in the boundaries of the charging object and small charging goals in the end charging map. The high-resolution images through remote sensing help in detecting the changes in the region more accurately than the existing method.

In 2023, Aniah *et al.* [30] have implemented remote sensing, Geographic Information System (GIS), and Markov chain model (MCM). Initially, MCM was used to depict the LULCC class categories where it contains the grid cells and each grid cell was fixed to its number of positions. Then MCM explained the transfer rate and quantification of conversion state in the various land-use types. Then the land-use data will be predicted on a large scale spending a shorter time.

In 2023, Pokhariya *et al.* [31] have used LANDSAT images. Initially, the images the classified into five LULC classes through the Random Forest classifier and Google Earth. They were classifying rural areas, urban areas, water regions, forest regions, agricultural areas, and empty lands. Then Kappa statistics was used to find the accuracy in the evaluation of the LULC.

In 2023, Lavanya *et al.* [32] have classified using the TSADL_LULCCD. The LANDSAT 7 images of the nallamalla were used. This method consists of three main approaches. Initially, the images from the satellite were fed into the Dense EfficientNet feature extraction. Then the various varieties of the land cover features were classified using the Deep Belief Network classifier (DBN). Finally, Tunicate Swarm Algorithm (TSA) was executed for the hyperparameter tuning. In the end, the experimental validation was done by comparing LANDSAT-7 images. This method can be used in any place similar to the nallamalla forest.

In 2023, Ahmad *et al.* [33] have implemented LANDSAT images. Initially, Landsat satellite images of Lahore were collected. Then cloud coverage was adjusted to a maximum of 15% in order to enhance the precision of the processed data. Then semi-automatic classification model was applied to obtain the images from other years. Then cellular automata (CA-ANN) model and modules of Land Use Change Evaluation (MOLUSCE) were implemented to stimulate the other use of land. Finally, from the

monitored data the change was detected and used as the result. However, this method was applicable only to the Lahore region.

In 2023, Xiang *et al.* [34] have experimented with DL with Sentinel-2 satellite data. Initially, the various DL model was compared along with the loss function to find the best-performing model. Then from the selected model, different metrics were assigned for evaluation, and changes in the land area and changes in the land area were predicted. Then the obtained images were compared with the sentinel-2 images along with the loss function to get the expected result. However, this process was not providing accurate mapping results.

In 2023, Affonso *et al.* [35] have executed a remote sensing tool called Volta Grande Do Xingi (VGX). Initially, the images are tested with two different methods. One is parametric and the other is non-parametric. Parametric has three methods and non-parametric respectively. Then the confusion matrix was used to perform to identify the evaluation function of the two methods. Furthermore, the post-classification comparison method was used to analyze the changes in the images. By comparing the overall classification, the overall accuracy was obtained. However certain factors like cloud cover, sensor noise, and atmospheric interference affect the accuracy of the result.

In 2023, Shimizu *et al.* [36] have experimented with an RF classifier and Landsat time series temporal segmentation for prediction. Initially, the early images of the location were collected and used Landsat data and LandTrendr temporal segmentation algorithm to obtain the key function of the land. They introduced separate random forest models to identify LULC and forest disturbances. With the result, they combined both LULC and Forest Disturbances as a map and find the coincided details over time. This data was merged with other classes to obtain accurate results however with the landscape's complexity and ephemeral nature the classification was visually difficult.

2.2 Challenges

- Due to the massive dependence in the segmentation model introduced to generate the objects, the object based detection mechanism poses a significant challenge. It was highly required to observe that the objects extracted through the segmentation model do not show same size. Hence, under and over segmentation errors degrades the accuracy in change detection [1].
- Even though various change detection methods are introduced, the target to lower the magnitude change in certain scenarios associated with the forest area poses a challenging issue in the remote sensing environment [7].
- Information gathering and change detection in forest are the time consuming and expensive studied with the integration of aerial and terrestrial surveys. However, to derive the basic parameters using the digital data analysis poses a major complex in forest area [4].
- The limitations arise in the change detection approaches were related to the class of input data, such that the various imaging geometries might degrade the accuracy and interferometric height of change estimate [5].
- The hyperspectral and the multispectral data do not hold any information to characterize the structure of forest. However, they are not used to perform the analysis in the geometric changes of trees [8].
- Due to the temporal changes in the land cover types with respect to time, selecting the training sets for various land cover class poses a challenging issue in forest region.

3. Proposed Crow Chicken Swarm optimization based deep Long Short Term Memory for forest change detection and prediction

This research predicts the changes in the forest area by detecting the accurate change measures. The forest change detection and prediction method is needed for identifying forest land cover change based on the high resolution satellite image. Due to the random variation in the sensor's response and the intrinsic characteristic features of land cover elements, the existing method fails to detect the changes accurately. Hence, a method is developed, which takes the advantages of low computational complexity and increases the computational efficiency. The proposed approach involves four different phases: pre-processing, vegetation indices, segmentation, and change detection, respectively. Initially, two input satellite images are collected at different time instances and are pre-processed separately using the Fuzzy Non-local means (Fuzzy NLM) filter. The resulted two different pre-processed images are fed to the vegetation indices phase, where the features, like RVI, EVI, GLAI, and CI-Green are effectively extracted from each images. The pixel value location of each image along with the extracted features is send to the segmentation phase, where the segmentation is carried out using the active contour approach. The segmented results from both the images are fed to the mapping module, where the

segment mapping for two images are performed based on the cosine similarity to detect the area as forest or non-forest. From the result of segmented mapping, the percentage of changes in the forest and non-forest area is detected using the pixel-based change detection and segment-based change detection. Then, the cumulative percentage of change detection is generated and is represented in the form of time series data, which is further used to achieve the percentage of change prediction. However, the percentage of changes in the forest area is predicted using the Deep LSTM classifier, which is trained by the proposed C-CSO algorithm. The proposed C-CSO algorithm is the integration of CSO [26] and CSA [27], respectively. Fig. 1 shows the schematic diagram of the proposed forest change prediction approach.

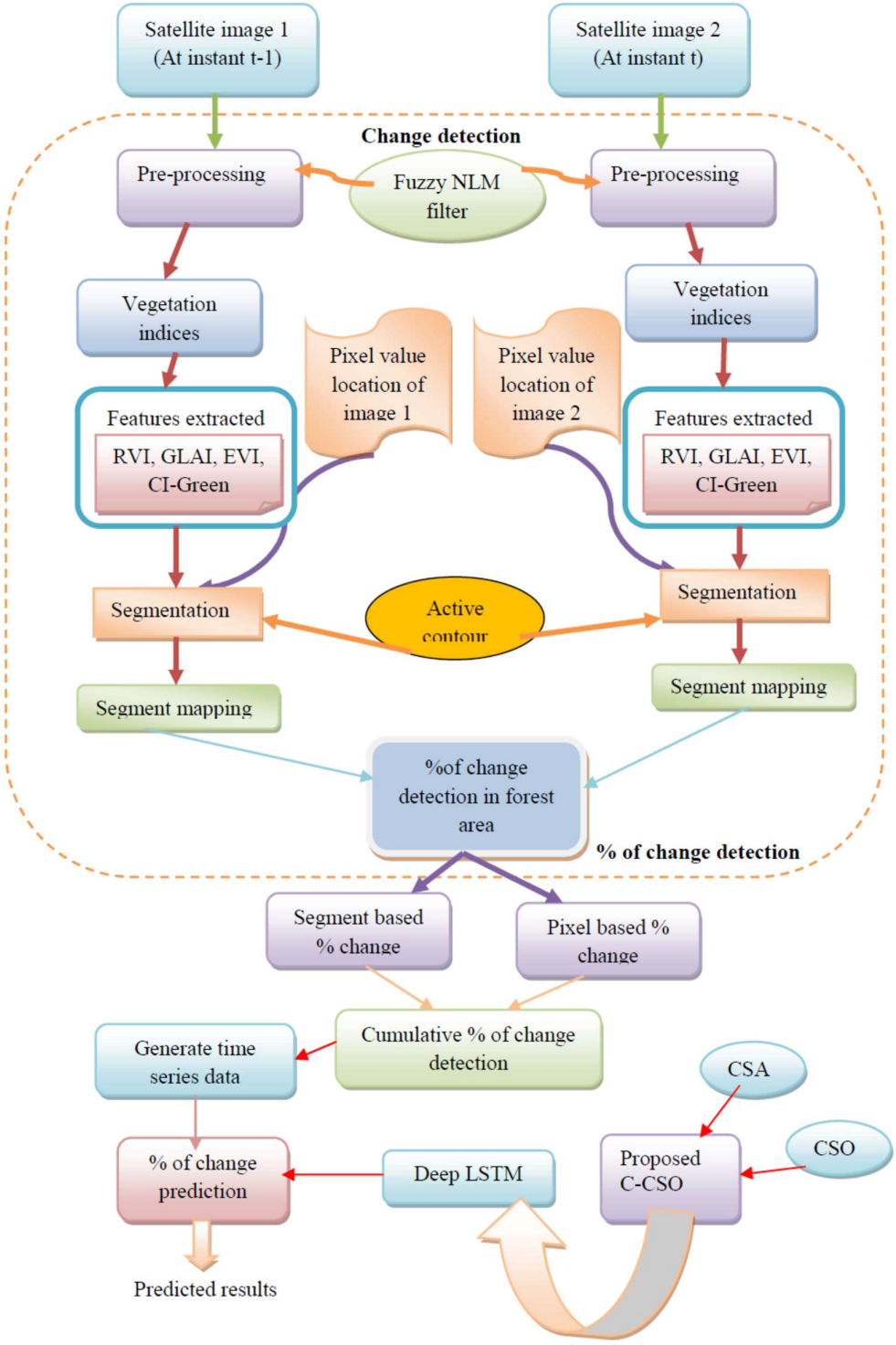


Fig.1. *Schematic diagram of the proposed forest change detection and prediction model using C-CSO based Deep LSTM*

3.1 Input Satellite Images

The input satellite image I is taken from the same location at two different time instances t and $t-1$, which is represented as, $\{I_t, I_{t-1}\}$ to perform the forest change detection and prediction. The two input satellite images are further subjected to the pre-processing stage in order to eliminate the noise in the images.

3.2 Pre-processing the Satellite Images

The input satellite images I_t and I_{t-1} are pre-processed using the Fuzzy NLM filter in order to remove the artifacts and noise present in the images. The Fuzzy NLM filter [23] identifies the non-local region from the images based on the fuzzy logic concept and finds the non-similar and similar regions situated around the noisy pixel. The non-local pixel values are identified based on the similarity factor for estimating the noise-free value. The major advantage of using the Fuzzy NLM filter is to select the homogeneous pixels from non-local neighbourhood region. The resulted pre-processed images are represented as, P_t and P_{t-1}, respectively.

3.3 Vegetation indices for Feature Extraction

The pre-processed images P_t and P_{t-1} are processed by the vegetation indices phase to extract the features, like RVI, GLAI, EVI, and CI-Green to perform the forest change detection and prediction. The above specified features are extracted from each individually pre-processed satellite images P_t and P_{t-1}. The vegetation indices are the transformation or the mathematical combination of spectral bands, which accentuates the properties of forests thus, they are distinct than other features. The vegetation information of the satellite imagery is obtained through the vegetation indices. Therefore, the vegetation indices obtained through the respective colour band is explained as follows:

Ratio Vegetation Index (RVI): RVI is also called as simple ratio (SR) vegetation index, which is used to specify the total amount of vegetation directly in the forest area. It is ratio of near infrared band to the red band of image, which is indicated as,

$$f_t^1 = \frac{A}{B} \tag{1}$$

where, A denotes the near infrared band, B denotes the red band, and f_t^1 represents the RVI features obtained from P_t image.

Enhanced Vegetation Index (EVI): EVI is considered as the highly-optimized index, which provides the spatial information of the biomass region. The EVI features is computed as,

$$f_t^2 = \frac{2.53(A-B)}{1+A+6\times B-7.50C} \tag{2}$$

where, f_t^3 denotes the EVI features of image P_t, and C denotes the blue band.

Green leaf area index (GLAI): The GLAI feature provides the information about biological characteristics in vegetation. It is defined as the ratio of leaf area to ground area, which is computed as,

$$f_t^3 = 3.618\times f_t^2 \tag{3}$$

where, f_t^3 denotes the GLAI feature obtained for image P_t.

Green Chlorophyll index (CI-Green): It is the most significant vegetation parameter, which offers the information concerning the physiological status of phenotypic manifestation. The CI-Green is represented as,

$$f_t^4 = \left(\frac{A}{D}\right) - 1 \tag{4}$$

where, D represents green band, and f_t^4 denotes the CI-Green feature for image P_t. Therefore, the resultant features extracted from the pre-processed image P_t is specified as,

$$f_t = \left\{f_t^1, f_t^2, f_t^3, f_t^4\right\} \tag{5}$$

Similarly, the above-mentioned features are extracted from the pre-processed image P_{t-1} and is also expressed as,

$$F_t = \left\{ F_t^1, F_t^2, F_t^3, F_t^4 \right\} \tag{6}$$

where, F_t^1 represents the RVI features obtained for P_{t-1} image, F_t^2 denotes the EVI features of image P_{t-1}, F_t^3 indicates GLAI feature for image P_{t-1}, and F_t^4 represents the CI-Green feature for image P_{t-1}, respectively. Hence, the features extracted from image P_t is denoted as, f_t and the features from image P_{t-1} is denoted as, F_{t-1}

3.4 Forest change Segmentation using Active Contour Approach

The segmentation process is required to ensure the effectiveness of forest change detection based on the pixel values of the pre-processed satellite images. Here, four extracted features along with the pixel value location of each satellite images are used to perform the segmentation process. The features extracted from each of the pre-processed image along with their pixel values are used to segment the images. Here, the segmentation of the images is carried out using the active contour model. The features f_t of image I_t along with their pixel value location, and the features F_{t-1} of image I_{t-1} along with their pixel value location are processed in the segmentation module separately based on the active contour model. Active contour model separates the image foreground from background image such that the region of interest from the image is further used to perform the detection processes. Active contour is used to segregate the pixels of interest from the satellite images. It defines the boundaries and specifies the area of interest to be required in the image. Active contour is a collection of pixel point, which undergoes the process of interpolation. The major advantage of using the active contour model is to define the smooth shape of image, and generates closed contour region. The segmented results obtained from the features f_t and f_{t-1} is denoted as, S_t and S_{t-1}, respectively.

3.5 Segment Mapping for Detecting the Changes in Forest Area

The segmented results S_t and S_{t-1} of the two satellite images I_t and I_{t-1} is fed to the segment mapping module, where the segment mapping is done based on the cosine similarity. Cosine similarity is the measure used to find the similarity between two segmented results in the product space. It is not a magnitude, but just the orientation to specify the cosine similarity value between two images. Based on the orientation of the segmented results, the value of the similarity is specified either as '1' or '2', respectively. The cosine similarity between the segmented results are computed as,

$$M = \frac{S_t . S_{t-1}}{|S_t \| S_{t-1}|} \tag{7}$$

where, M denotes the similarity measure between the two images. From the segmented output of two images, the segment mapping is performed for change detection based on the cosine similarity. The segment mapping module mapped the two segmented results to find the forest and non-forest area and indicates the forest area as '1', and non-forest area as '2', respectively. Therefore, the output obtained from the segmented mapping module for the two segmented results as denoted as, M_t and M_{t-1}.

3.6 Compute the Percentage of Change Detection

Once the forest and the non-forest regions are identified then, it is required to compute the percentage of change detection using the pixel-based change detection and segment-based change detection. From the pixel-based and the segment-based change detection value, the cumulative percentage of change detection is calculated. The pixel-based change detection of the two mapped images M_t and M_{t-1} is shown in fig.2. Let us consider that the forest regions are placed with the pixel value of '1' and the non-forest regions are placed with the pixel value of '2'. From this, the percentage of pixel-based change detection in the forest area is computed by mapping the two images M_t and M_{t-1}.

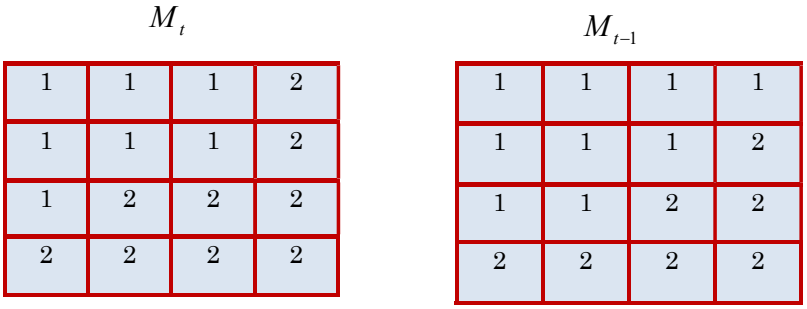


Fig.2. *Pixel-based change Detection of Forest and Non-Forest Area*

The forest area covered in both the images is considered to compute the percentage of change detection. As the pixel value '1' specifies the forest area in both the images M_t and M_{t-1}, the percentage of pixel based change detection in the forest area is computed as,

$$CD_{pixel} = \frac{9-7}{9} \times 100 \tag{8}$$

$$CD_{pixel} = 22.22\% \tag{9}$$

where, CD_{pixel} denotes the percentage of change detection in the forest area.

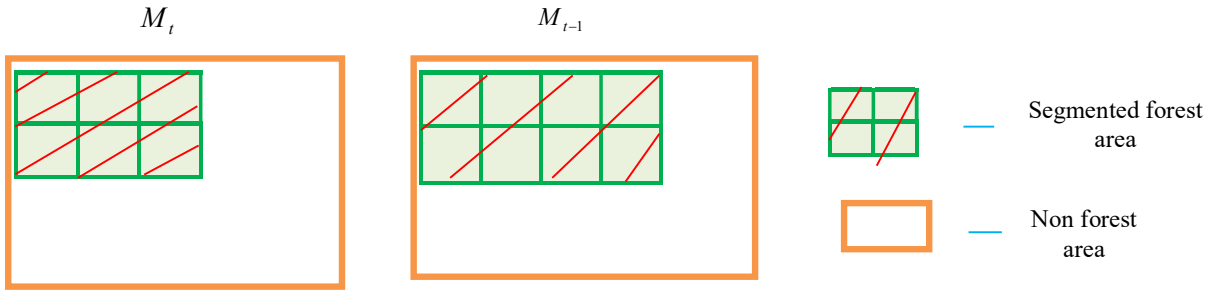


Fig.3.*Segment-based change Detection of Forest and Non-Forest Area*

The segment based change detection of the two mapped images M_t and M_{t-1} is shown in fig. 3. The percentage of segment based change detection is computed by considering the segmented forest area in the segmented results. Here, the shaded region specifies the segmented forest area, while the un-shaded portion represents the non-forest area. By mapping the two images M_t and M_{t-1}, the percentage of segment based change detection is computed as,

$$CD_{segment} = \frac{8-6}{8} \times 100 \tag{10}$$

$$CD_{segment} = 25\% \tag{11}$$

where, $CD_{segment}$ represents the percentage of segment-based change detection in the forest area. By taking the value of pixel-based and segment-based change detection, the cumulative percentage of change detection is computed. The cumulative percentage of forest change detection is calculated as,

$$CD_{forest} = \frac{22.22 + 25}{2} \tag{12}$$

$$CD_{forest} = 23.61\% \tag{13}$$

where, CD_{forest} denotes the cumulative percentage of change detection.

3.7 Compute the Percentage of Change Prediction

To compute the percentage of change prediction, it is required to generate the time-series data based on the output of percentage of change detection. The time-series data is generated from the pixel values of the cumulative percentage of change detection. The time-series data is generated from the output of percentage of change detection at time $\{t, t-1, t-2, t-3, ...\delta\}$. The time series data is fed as the input to the Deep LSTM classifier to compute the percentage of change prediction, as the Deep LSTM classifier effectively performs the prediction mechanism only based on the time series data.

3.7.1 Deep Long Short Term Memory Classifier

The time series data generated based on the output of percentage of change detection is fed as the input to the Deep LSTM classifier. The Deep LSTM classifier consists of convolutional structures in the state to state and input to state transition. The Deep LSTM is structured with the input $\{t, t-1, t-2, t-3, \ldots \delta\}$, output $\{G_1, \ldots G_\delta\}$, hidden state $\{K_1, \ldots . K_\delta\}$, and gates V_δ, T_δ and W_δ, respectively. Let us assume the input as $\{Z_1, \ldots \ldots Z_\delta\}$ instead of using the format $\{t, t-1, t-2, t-3, \ldots \delta\}$.The Deep LSTM [24] predicts the future based on the input and the past values that corresponding to the neighboring state, and the prediction is performed using the Hadamard product $\circ$ and convolutional operator $*$ .The Hadamard product is used to ensure the nature of cells, and the Deep LSTM yields the feature patterns based on the transitional kernel. The architecture of the Deep LSTM is organized with the forecasting and encoding layers. The output of the cell and the initial states of the encoding network is fed to the forecasting network. Fig. 4 illustrates the architecture of the Deep LSTM classifier.

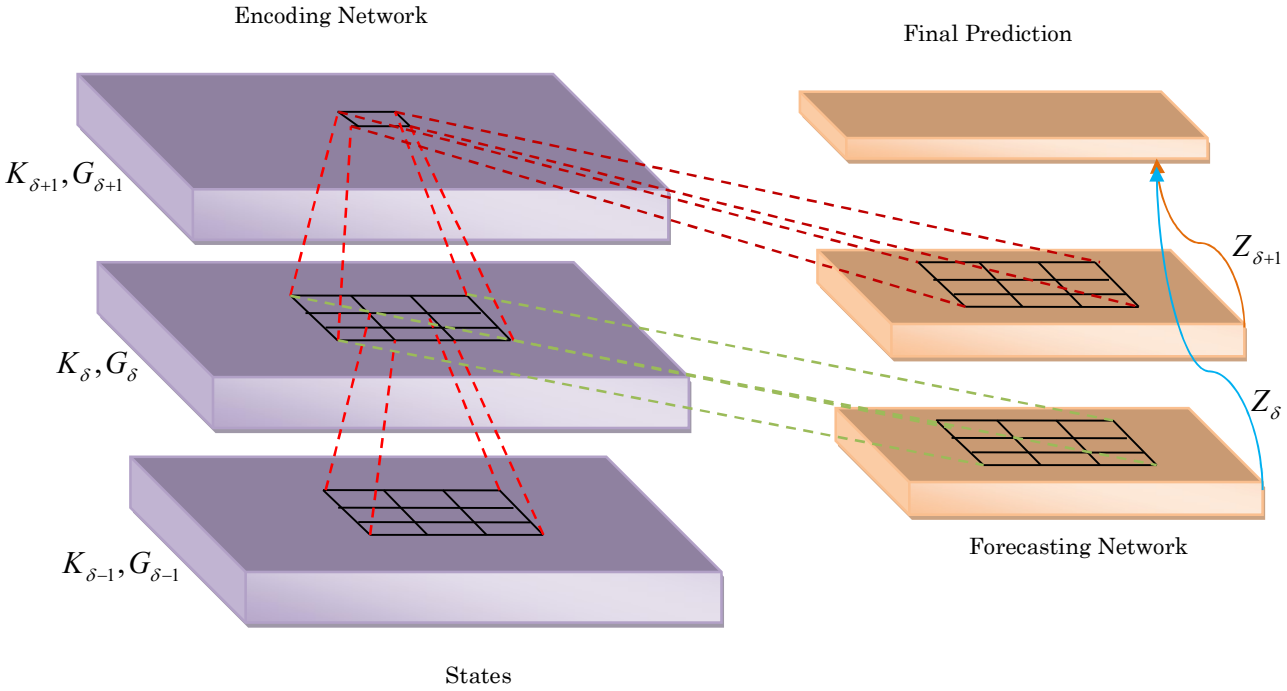


Fig.4. *Architecture of the Deep LSTM classifier*

The Deep LSTM compresses the input value to the hidden states for forecasting the forest change prediction. Each network consists of the memory units that hold the memory cell and gates. The gates and the memory cell of the Deep LSTM classifier plays a key role to control the information flow based on the spatio temporal sequences. The memory state of the Deep LSTM is modified with the input, and forget gate. The Deep LSTM classifier is more effective and convenient to perform the forest change prediction based on the time series data. The output received from the input gate is expressed as,

$$V_\delta = \lambda\left(w_V^Z * Z_\delta + w_V^K * K_{\delta-1} + w_V^G \circ G_{\delta-1} + \mu^V \right) \tag{14}$$

where, Z_δ denotes the input vector, w_V^Z denotes the weigh between input layer and the input gate, λ represents the gate activation function, w_V^K indicates the weight between input layer and memory output, and w_V^G denotes the weight vector between input layer and cell output. $K_{\delta-1}$ and $G_{\delta-1}$ represents the previous output of cell and the memory unit, and μ^V denotes the bias of input layer. $\circ$ is the element wise multiplication, and $*$ denotes the convolutional operator. The output obtained from the forget gate is represented as,

$$T_\delta = \lambda\left(w_T^Z * Z_\delta + w_T^K * K_{\delta-1} + w_T^G \circ G_{\delta-1} + \mu^T \right) \tag{15}$$

where, w_T^Z denotes the weigh between input layer and the forget gate, w_T^K denotes the weight between the output gate and memory unit, and w_T^G denotes the weight between output gate and cell. μ^T represents the bias of forget gate. The output obtained from the output gate is represented as,

$$W_\delta = \lambda\left(w_W^Z * Z_\delta + w_W^K * K_{\delta-1} + w_W^G \circ G_{\delta-1} + \mu^W \right) \tag{16}$$

where, w_W^Z represents the weight between output gate and input layer, w_W^K indicates the weight between output gate and memory unit, w_W^G denotes the weight between the output gate and cell, and μ^W is the bias of output gate. Based on the activation function that corresponds to the cell, the output of the cell state is represented as,

$$\tilde{G}_\delta = \tanh\left(w_E^Z * Z_\delta + w_E^K * K_{\delta-1} + \mu^E \right) \tag{17}$$

where, w_E^Z denotes the weight between cell and input layer, w_E^K is the weight between cell and memory unit, and μ^E denotes the bias of cell.

The output of the cell is specified as the sum of the temporary cell, and the difference among the memory unit of previous and current layer is expressed as,

$$G_\delta = T_\delta \circ G_{\delta-1} + V_\delta \circ \tilde{G}_\delta \tag{18}$$

$$G_\delta = T_\delta \circ G_{\delta-1} + V_\delta \circ \tanh\left(w_E^Z * Z_\delta + w_E^K * K_{\delta-1} + \mu^E \right) \tag{19}$$

The output obtained from the memory unit is specified as,

$$K_\delta = W_\delta \circ \tanh(G_\delta) \tag{20}$$

where, K_δ represents the memory block output, and W_δ indicates the output gate. Therefore, the output generated from the output layer is represented as,

$$Y_\delta = \varphi\left(w_Y^K . K_\delta + \mu^Y \right) \tag{21}$$

where, Y_δ represents the output vector, w_Y^K represents the weight between output vector and memory unit, and μ^Y denotes the bias of output layer, respectively. The Deep LSTM classifier ensure the effectiveness of prediction in the forest change based on the time series data, such that the Deep LSTM is trained by the proposed C-CSO algorithm to compute the percentage of change prediction.

3.8 Proposed Crow Chicken Swarm Optimization Algorithm for Forest Change Prediction

The proposed C-CSO algorithm is used to train the Deep LSTM classifier to ensure the effectiveness of prediction performance. The proposed C-CSO algorithm is the integration of Chicken swarm optimization (CSO) [26] and Crow search algorithm (CSA) [27]. CSA is a meta-heuristic algorithm that provides the balance between the intensification and diversification. The crows are considered as the most intelligent birds, as they have largest brain in relative to the body size of crows. The crows have the self awareness features in mirror test and also have the ability to make the tool. Crows can remember the faces and warn other crows if any unfriend one approaches. Moreover, the crows can communicate and recall the food source of hidden location. The key principles of the crow search optimization algorithm are defined as follows:

- The crows live in the form of flocks
- It memorizes the location of hidden places.
- The crows updated their location by following other crows to forage the food.
- They protect the food source of themselves from being pilfered.

In the CSA algorithm, the intensification and the diversification are controlled using the parameter named awareness probability. However, the CSA conducts the search in the local region, such that the better solution is found in the local region. When the value of awareness probability increases, the probability to search the good solution reduces, hence the CSA is explored in the global scale.

Fitness function: The fitness function is computed to identify the best solution with the optimal fitness value based on the updated position of crow. The solution with the minimal error value is accepted as the best solution. The fitness function is computed as,

$$N_{fit} = \frac{1}{b}\sum_{h=1}^{b} Y_{\delta} - \mu \tag{22}$$

where, Y_{δ} denotes the output of classifier, μ denotes the estimated output, and N_{fit} represents the fitness value. The algorithmic steps involved in the proposed C-CSO algorithm are explained as follows:

i) Population initialization in search space: The population of the crows are initialized with U number of crows in the n-dimensional environment, flight length as L, and Q as the awareness probability. The crow m at iteration r in the search space is represented as,

$$X_m^r(m = 1,2,...U) = \left\{X_1^r, X_2^r, ...X_U^r\right\};\ \{r = 1,2,...r_{max} \tag{23}$$

where, r is the iteration, and r_{max} denotes the maximum iteration.

ii) Initialize the memory and location of crow: The U number of crows is randomly positioned in the n dimensional search space. Each crow specifies the feasible solution and a denotes the decision variable.

$$\alpha = \begin{bmatrix} X_{1,n}^1 & X_{2,n}^1 & ... & X_{\tau,n}^1 \\ X_{2,n}^2 & X_{2,n}^2 & ... & X_{\tau,n}^2 \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ X_{1,n}^{\upsilon} & X_{2,n}^{\upsilon} & ... & X_{\tau,n}^{\upsilon} \end{bmatrix} \tag{24}$$

The memory of the crows is initialized, such that the crows have no experience at the initial iteration, it is considered that the crow hides the food at the initial position.

$$\beta = \begin{bmatrix} R_{1,n}^1 & R_{2,n}^1 & ... & R_{\tau,n}^1 \\ R_{2,n}^2 & R_{2,n}^2 & ... & R_{\tau,n}^2 \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ R_{1,n}^{\upsilon} & R_{2,n}^{\upsilon} & ... & R_{\tau,n}^{\upsilon} \end{bmatrix} \tag{25}$$

where, α denotes the position of the crow, and β represents the memory of the crow.

iii) Update the position of crow based on the food sources: Each crow contains the memory in which the location of hiding place is memorized. The crows move in the search space to search the food sources. The crow m decides to follow the crow k to know the hiding location of food source of crow K. The crow k does not know that the crow m is following it, Therefore, the crow m move towards the hidden location of crow k, hence the new updated position of crow m is represented as,

$$X_{m,n}^{r+1} = X_{m,n}^r + l_1 \times L_{m,n}^r \times \left(a_k^r - X_{m,n}^r\right) \tag{26}$$

where, l_1 denotes the random number that lies between 0 and 1, $L_{m,n}^r$ represents the flight length of crow m.

$$X_{m,n}^{r+1} = X_{m,n}^r\left(1 - l_1 L_{m,n}^r\right) + l_1 L_{m,n}^r a_k^r \tag{27}$$

The above equation is the updated equation of CSA algorithm which is modified with the updated equation of CSO algorithm.

The updated equation of movement of hen towards the food source in the rooster groups is represented as,

$$X_{m,n}^{r+1} = X_{m,n}^r + p_1 * z\left(X_{ll,n}^{r,} - X_{m,n}^r\right) + p_2 * z\left(X_{l2,n}^r - X_{m,n}^r\right) \tag{28}$$

where, z is the random number lies between o and 1. r denotes the iteration, p_1 and P_2 are the of rooster parameters.

$$X_{m,n}^{r+1} = X_{m,n}^r\left(1 - p_1 z \quad - p_2 z \quad\right) + p_1 z X_{ll,n}^{r,} + p_2 z X_{l2,n}^r \tag{29}$$

$$X_{m,n}^{r} = \frac{1}{1 - p_1 z \quad - p_2 z}\left(X_{m,n}^{r+1} - p_1 z X_{ll,n}^{r,} - p_2 z X_{l2,n}^{r} \right) \tag{30}$$

The above Eq. (30) is the updated equation of hen movement in the CSO algorithm. By substituting the Eq. (30) in Eq. (27), the resulted modified equation of the proposed C-CSO algorithm is expressed as,

$$X_{m,n}^{r+1} = \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z}\left(X_{m,n}^{r+1} - p_1 z X_{ll,n}^{r,} - p_2 z X_{l2,n}^{r} \right) + l_1 L_{m,n}^{r} a_k^{r} \tag{31}$$

$$X_{m,n}^{r+1} = \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z} X_{m,n}^{r+1} + \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z}\left(- p_1 z X_{ll,n}^{r,} - p_2 z X_{l2,n}^{r} \right) + l_1 L_{m,n}^{r} a_k^{r} \tag{32}$$

$$X_{m,n}^{r+1} - \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z} X_{m,n}^{r+1} = l_1 L_{m,n}^{r} a_k^{r} - \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z}\left(p_1 z X_{ll,n}^{r,} + p_2 z X_{l2,n}^{r} \right) \tag{33}$$

$$X_{m,n}^{r+1}\left(1 - - \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z} \right) = l_1 L_{m,n}^{r} a_k^{r} - \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z}\left(p_1 z X_{ll,n}^{r,} + p_2 z X_{l2,n}^{r} \right) \tag{34}$$

$$X_{m,n}^{r+1}\left(\frac{1 - p_1 z \quad - p_2 z - 1 + l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z} \right) = l_1 L_{m,n}^{r} a_k^{r} - \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z}\left(p_1 z X_{ll,n}^{r,} + p_2 z X_{l2,n}^{r} \right) \tag{35}$$

$$X_{m,n}^{r+1}\left(\frac{p_1 z \quad - p_2 z + l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z} \right) = l_1 L_{m,n}^{r} a_k^{r} - \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z}\left(p_1 z X_{ll,n}^{r,} + p_2 z X_{l2,n}^{r} \right) \tag{36}$$

$$X_{m,n}^{r+1} = \frac{1 - p_1 z - p_2 z}{l_1 L_{m,n}^{r} - z(p_1 + p_2)}\left\{ l_1 L_{m,n}^{r} a_k^{r} - \frac{1 - l_1 L_{m,n}^{r}}{1 - p_1 z \quad - p_2 z}\left(p_1 z X_{ll,n}^{r,} + p_2 z X_{l2,n}^{r} \right) \right\} \tag{37}$$

where, z is the random number that lies between $[0,1]$, p_1 and p_2 are the rooster parameters, that are specified as,

$$p_1 = \exp\left(\frac{(b - b_{r1})}{abs(b) + \eta} \right) \tag{38}$$

$$P_2 = \exp(b_{r2} - b) \tag{39}$$

Here, $r1$ and $r2$ are the random number that lies between 0 and 1, and b is the fitness of chickens.

The above Eq. (36) defines the final updated equation of the proposed C-CSO algorithm, which ensures the effective of the proposed algorithm to perform the forest change prediction.

iv) Evaluate the feasibility: The fitness function is computed using the Eq. (22), the function with the minimal error value is taken as the best optimal solution.

v) Generate the new position: The crow generates a new position by considering one of the flock crow k and follow it to generate the new position to know the hidden location of food source.

vi) Update the memory: The crow updates their memory based on the fitness value, if the new position of the crow have better fitness value, than the memorized position, then the crow updates the memory with respect to the new position.

vii) Termination: The above steps are repeated until the best solution is obtained or satisfies the condition criteria. Algorithm 1 shows the pseudo code of the proposed C-CSO based Deep LSTM classifier.

Algorithm 1. Pseudo code of the proposed C-CSO based LSTM

Sl. no	Pseudo code of the proposed C-CSO based Deep LSTM
1	Initialize the population
2	Specify the memory of crow
3	Evaluate the position
4	while $r < r_{\max}$
5	for $m = 1,....U$
6	Randomly select one crow (like n)to follow
7	Specify the awareness probability Q
8	if $l_q \geq Q$ l_q-random number
9	Update the position using Eq. (37)
10	else
11	Select a random position
12	end if
13	end for
14	Evaluate the feasibility
15	Compute the new position
16	Update the memory
17	end while

The intelligent behaviour of the crow is incorporated with the hierarchical order and the mimicking behaviour of chickens in order to enhance the effectiveness of change detection in the forest area. The parametric features from the crow optimization provides better prediction results, but to increase the effectiveness of performance, the behaviour of chicken swarm is integrated with the crow optimization. Hence, the proposed C-CSO based Deep LSTM computes the percentage of change prediction using the Deep LSTM classifier, which is trained by the proposed meta-heuristic optimization named C-CSO algorithm.

4. Results and Discussion

The results and discussion made using the proposed C-CSO based Deep LSTM classifier based on the evaluation metrics, like MSE, accuracy, and prediction error is explained in this section.

4.1 Experimental Setup

The implementation of the proposed C-CSO based Deep LSTM classifier is carried out in the MATLAB tool using the dataset named Global PALSAR-2/PALSAR/JERS-1 Mosaic and Forest/Non-forest Map [25]. This dataset contains numerous satellite images used to respond and understand the global environmental changes, such as biodiversity loss, and global warming. It observes the surface of lands even under cloud, and therefore this dataset provides useful information regarding the forest changes in the tropical region.

4.2 Evaluation Metrics

The performance of the proposed C-CSO based Deep LSTM classifier is evaluated using the metrics, like MSE, accuracy, and prediction error, respectively.

Mean Square Error (MSE): It is the measure that specifies the average squares of error, which is represented as,

$$x = \frac{1}{p}\sum_{\delta=1}^{p}\left(Y_\delta - \hat{Y}_\delta\right)^2 \tag{40}$$

where, x is the MSE, Y_δ is the error value, and p denotes the predictions, respectively.

Accuracy: It is a measure used to accurately predict the changes in the forest are, which is computed as,

$$A = \frac{a+b}{a+b+c+d} \tag{41}$$

where, a is the true positive rate, b is the true negative, c is the false positive and d denotes false negative, A represent accuracy.

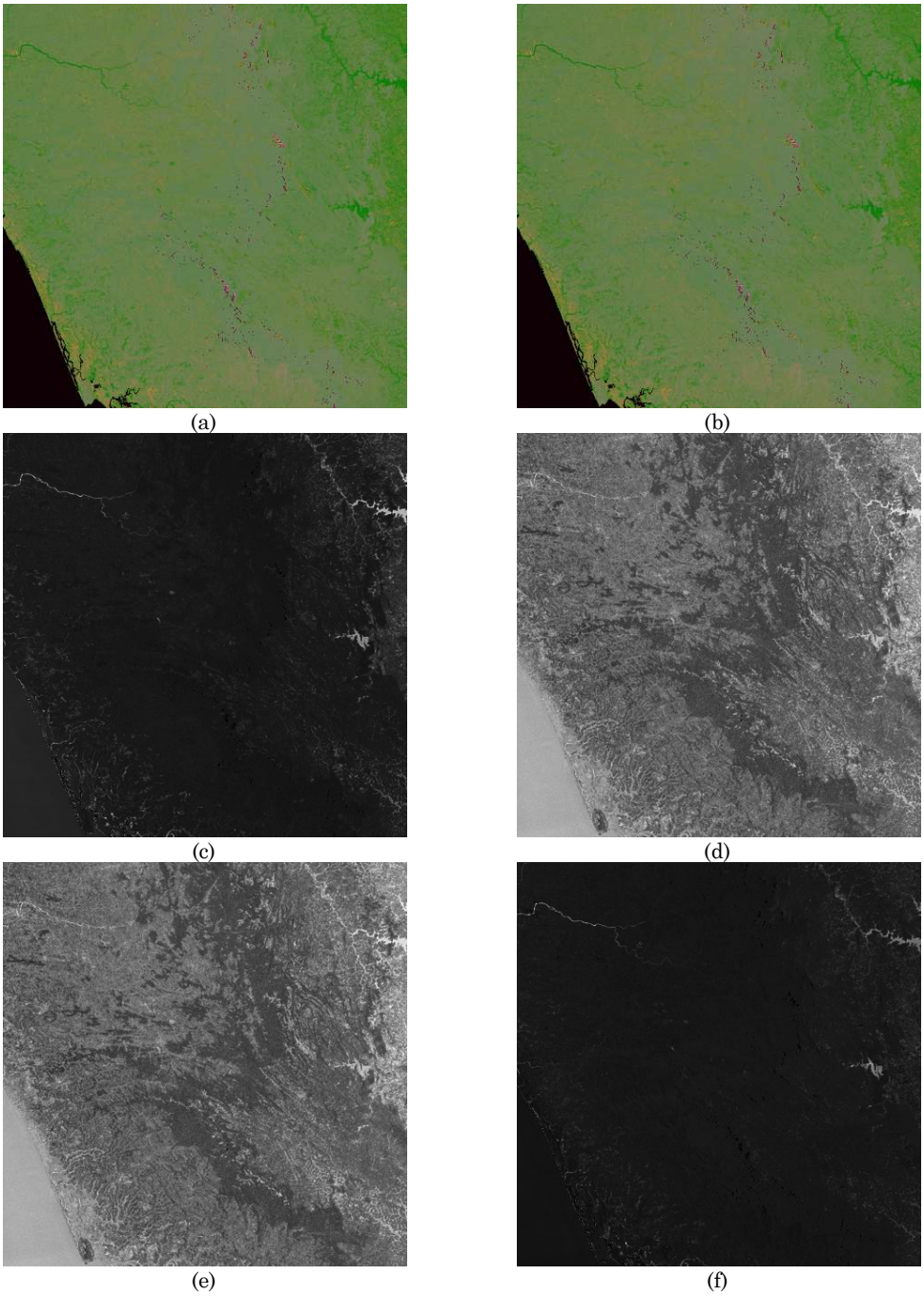
Prediction error: It is the ratio of different between the measured value and the predicted value to the total measured value, which is expressed as,

$$P = \frac{\alpha - \beta}{\alpha} \tag{42}$$

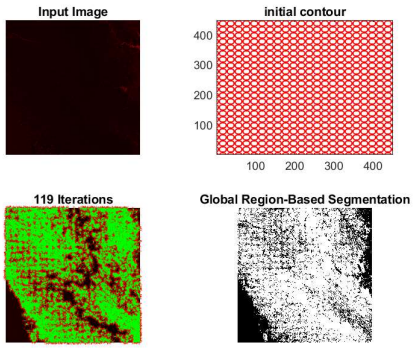
where, α denotes the measure value, and β is the predicted value.

4.3 Experimental Results

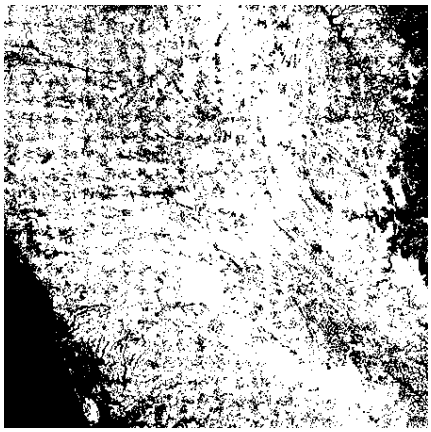
The experimental results of the proposed forest change detection and prediction approach is discussed in this section. Fig.5 show the experimental results for forest image-1, and fig. 5 a) shows the input forest image-1. Fig. 5 b) represents the pre-processed image-1, and fig. 5 c) portrays the extracted RVI features for image-1. Fig. 5 d) shows the extracted GLAI features for image-1, and fig.5 e) represents the extracted EVI features for image-1. Fig. 5 f) shows the extracted CI-Green features for image-1, fig. 5 g) shows the active contour process for image-1, and fig. 5 h) depicts segmented result for image-1.



(a) (b) (c) (d) (e) (f)



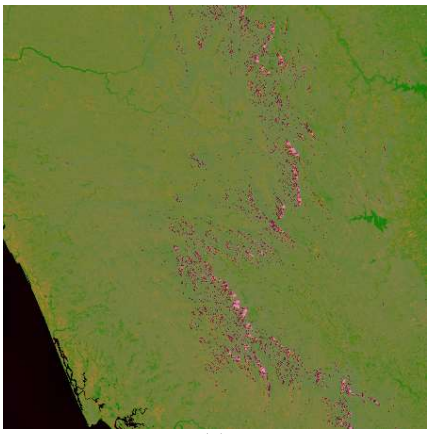
(g)



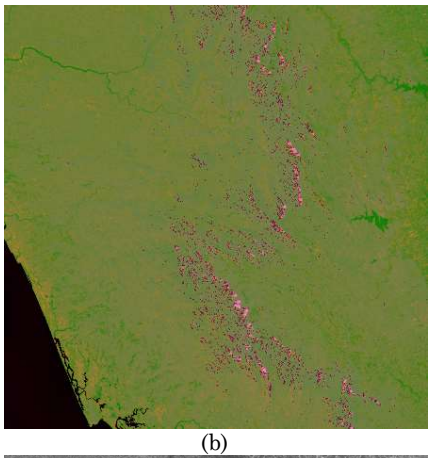
(h)

Fig.5. *Experimental results of forest image-1, a) input forest image-1, b) pre-processed image-1, c) extracted RVI features for image-1, d) extracted GLAI features for image-1, e) extracted EVI features for image-1, f) extracted CI-Green features for image-1, g) active contour process for image-1, h) segmented result for image-1.*

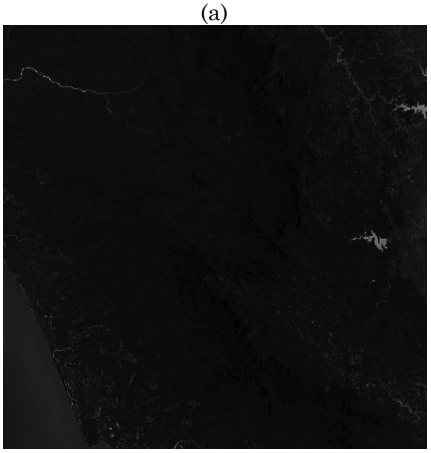
Fig. 6 show the experimental results for forest image-2, and figure 6 a) shows the input forest image-2. Fig. 6 b) represents the pre-processed image-2, and figure 6 c) portrays the extracted RVI features for image-2. Fig. 6 d) shows the extracted GLAI features for image-2, and fig.6 e) represents the extracted EVI features for image-2. Fig. 6 f) shows the extracted CI-Green features for image-2, fig. 6 g) shows the active contour process for image-2, and fig. 6 h) depicts segmented result for image-2.



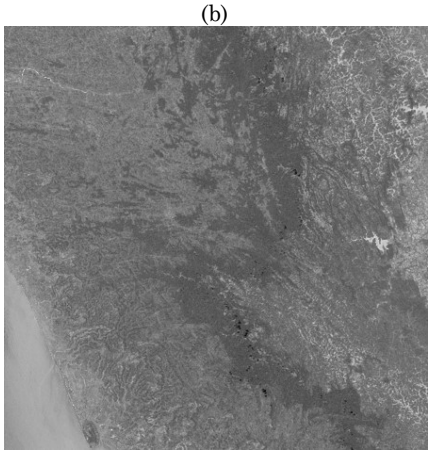
(a)



(b)



(c)



(d)

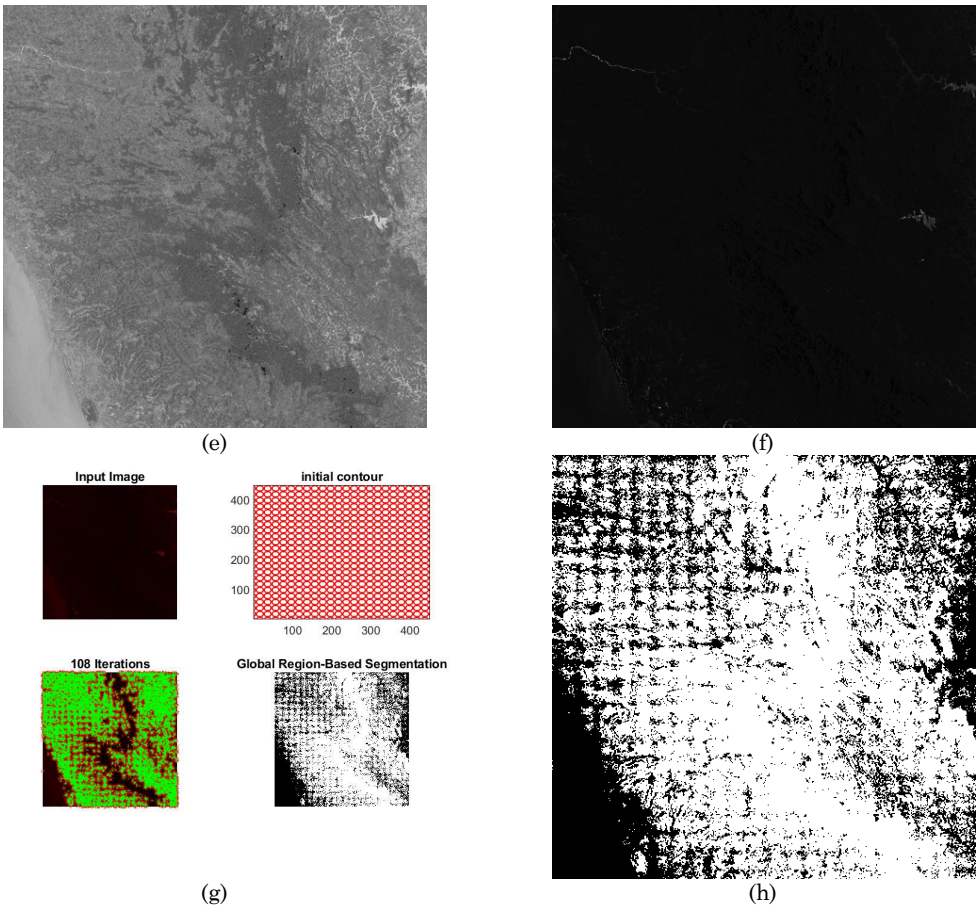


Fig.6. *Experimental results of forest image-2, a) input forest image-2, b) pre-processed image-2, c) extracted RVI features for image-2, d) extracted GLAI features for image-2, e) extracted EVI features for image-2, f) extracted CI-Green features for image-2, g) active contour process for image-2, h) segmented result for image-2.*

4.4 Performance Analysis

The performance analysis made using the proposed approach for forest change detection and prediction is elaborated in this section.

4.4.1 Analysis using Four Features

The performance analysis made using the four features, like RVI, GLAI, EVI, and CI-Green is explained in this section. Fig. 7 shows the analysis made using four features. Fig.7 a) shows the analysis of detected change. In the range of 2009-2010 years, the detected change observed from the forest image using the four features is 3.6062%. Fig. 7 b) shows the analysis of accuracy. In the year of 2010 the segmentation accuracy observed from the forest image data using the four features is 0.9024. Fig. 7 c) represents the analysis of error. In the year of 2010 the segmentation error observed from the forest image data using the four features is 0.18088.

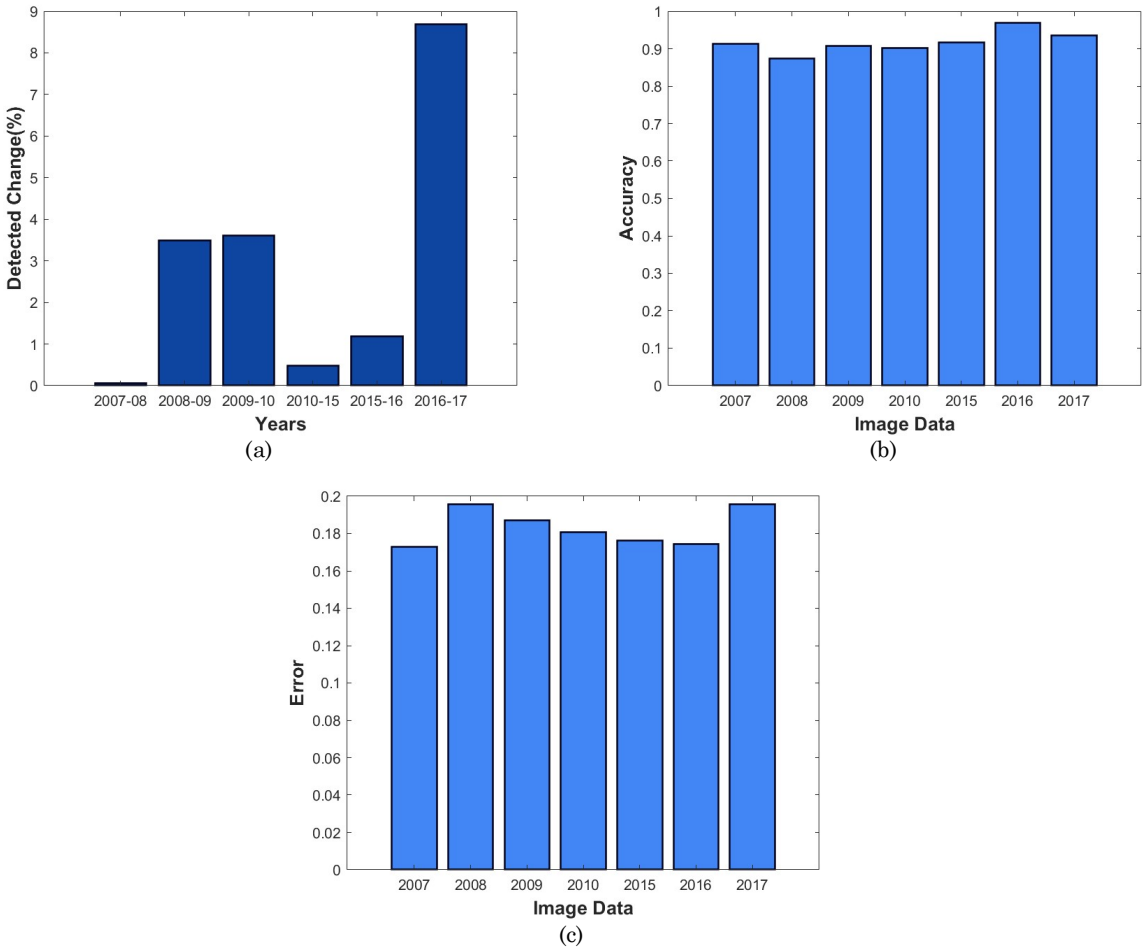
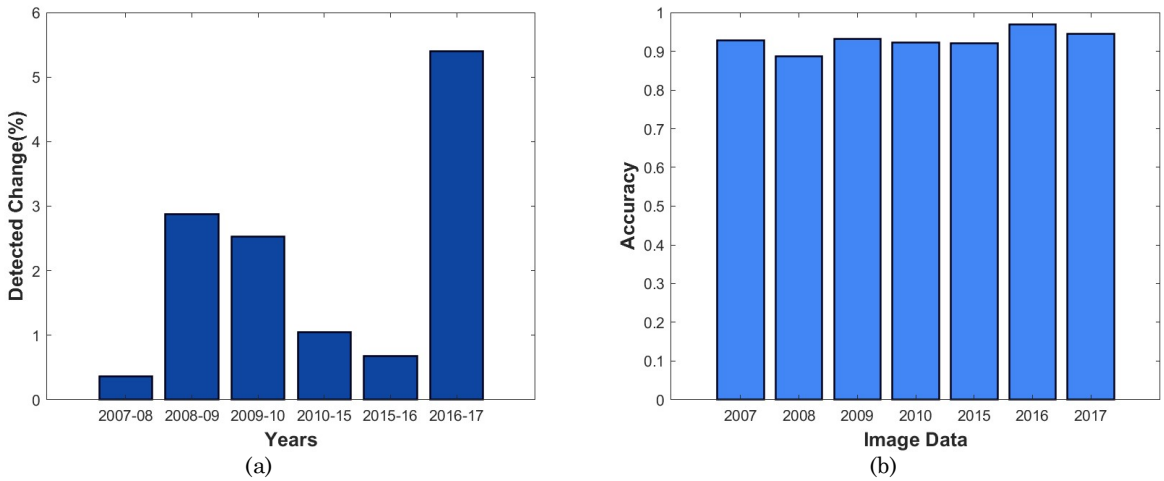
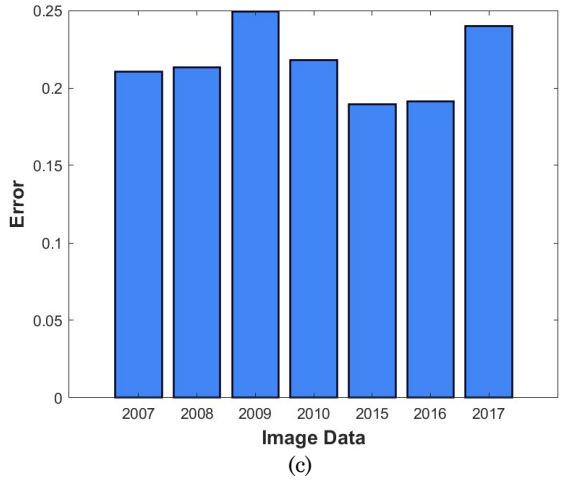


Fig.7. *Analysis using four features, a) detected change, b) accuracy, c) error*

4.4.2 Analysis using Two Features

The performance analysis made using two features, namely RVI, and GLAI is explained in this section. Fig. 8 shows the analysis made using two features. Fig. 8 a) shows the analysis of detected change. In the range of 2009-2010 years, the detected change observed from the forest image using two features is 2.5313%. Fig. 8 b) shows the analysis of accuracy. In the year of 2010 the segmentation accuracy observed from the forest image data using two features is 0.9229. Fig. 8 c) represents the analysis of error. In the year of 2010 the segmentation error observed from the forest image data using two features is 0.2178.



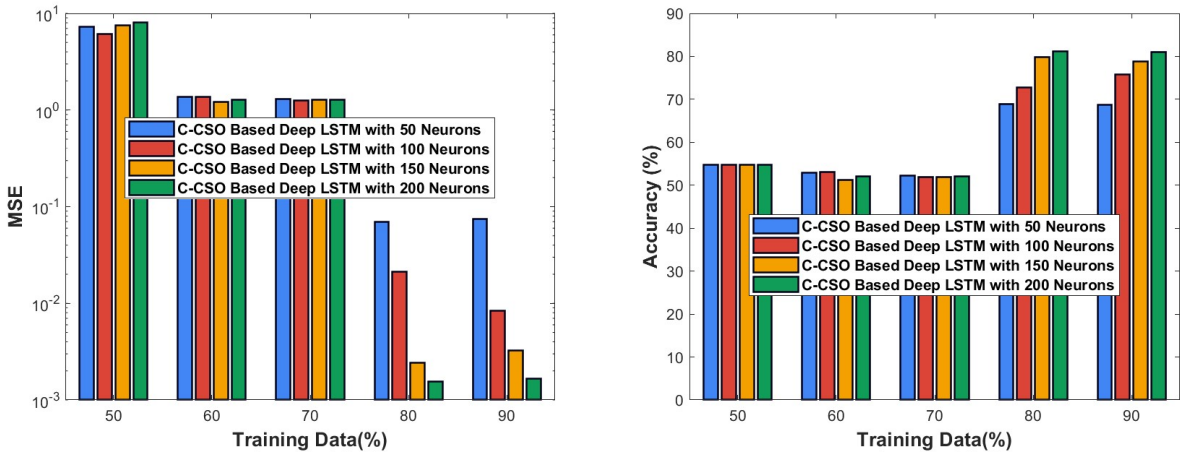


(c)

Fig.8. *Analysis using two features, a) detected change, b) accuracy, c) error*

4.4.3 Performance Analysis by Varying Neurons

Fig. 9 shows the performance analysis of the proposed approach by varying the neurons. Fig. 9 a) depicts the analysis of MSE by varying the training data. When training data=70%, the MSE obtained by the proposed C-CSO based Deep LSTM with 50 neurons, C-CSO based Deep LSTM with 100 neurons, C-CSO based Deep LSTM with 150 neurons, and C-CSO based Deep LSTM with 200 neurons is 1.2929, 1.2644, 1.2705, and 1.2760, respectively. Fig. 9 b) depicts the analysis of accuracy by varying the training data. When training data=80%, the accuracy obtained by the proposed C-CSO based Deep LSTM with 50 neurons, C-CSO based Deep LSTM with 100 neurons, C-CSO based Deep LSTM with 150 neurons, and C-CSO based Deep LSTM with 200 neurons is 68.944%, 72.825%, 79.811%, and 81.2%, respectively.

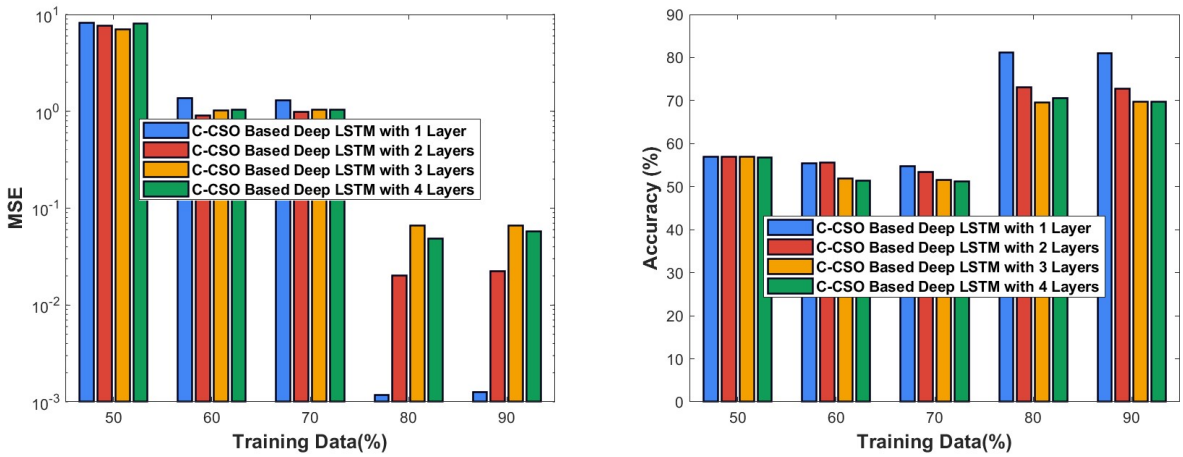


(a) (b)

Fig.9. *Performance analysis by varying neurons, a) MSE, b) accuracy*

4.4.4 Performance Analysis by Varying Layers

Fig. 10 shows the performance analysis of the proposed approach by varying the neurons. Fig. 10 a) represents the performance analysis of MSE by varying the training data. When training data=70%, the MSE obtained by the proposed C-CSO based Deep LSTM with 1 layer, C-CSO based Deep LSTM with 2 layers, C-CSO based Deep LSTM with 3 layers, and C-CSO based Deep LSTM with 4 layers is 1.2978, 0.9940, 1.0331, and 1.0395, respectively. Fig. 10 b) represents the performance analysis of accuracy by varying the training data. When training data=70%, the accuracy obtained by the proposed C-CSO based Deep LSTM with 1 layer, C-CSO based Deep LSTM with 2 layers, C-CSO based Deep LSTM with 3 layers, and C-CSO based Deep LSTM with 4 layers is 54.803%, 53.336%, 51.610%, and 51.200%, respectively.



(a) (b)

Fig.10. *Performance analysis by varying layers, a) MSE, b) accuracy*

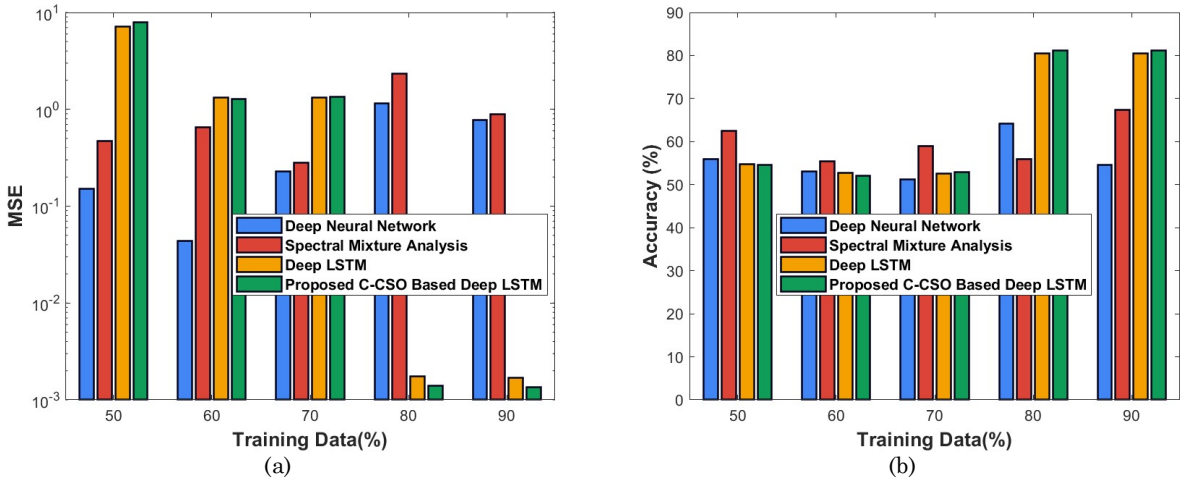
4.5 Comparative Methods

The performance improvement obtained by the proposed method is revealed by comparing the proposed with the existing methods, like deep neural network [1], spectral mixture analysis [2], Deep Long Short Term Memory (Deep LSTM) [28], respectively.

4.6 Comparative Analysis

The comparative analysis made using the proposed approach with respect to the existing methods is discussed in this section. Fig. 11 shows the comparative analysis of the proposed approach. Fig. 11 a) depicts the comparative analysis of MSE with respect to the training data. When training data=60%, the MSE obtained by the existing methods, like deep neural network, spectral mixture analysis, and Deep LSTM is 0.0438, 0.6495, and 1.3335, while the proposed C-CSO based Deep LSTM obtained lower MSE of 1.2819, respectively.

Fig. 11 b) depicts the comparative analysis of accuracy with respect to the training data. When training data=60%, the accuracy obtained by the existing methods, like deep neural network, spectral mixture analysis, and Deep LSTM is 64.259%, 55.928%, and 80.434%, while the proposed C-CSO based Deep LSTM obtained better accuracy of 81.101%, respectively.



(a) (b)

Fig.11. *Comparative analysis, a) MSE, b) accuracy*

4.7 Comparative Discussion

Table 1 shows the comparative discussion of the proposed C-CSO based Deep LSTM classifier. From the below table it is clearly specified that the proposed classifier obtained better performance than the existing literature works. The accuracy obtained by the existing methods, like deep neural network,

spectral mixture analysis, and Deep LSTM is 54.616%, 67.315%, and 80.553%, while the proposed obtained better accuracy of 81.200% for 90% training data. The MSE obtained by the existing methods, like deep neural network, spectral mixture analysis, and Deep LSTM is 0.7797, 0.8882, and 0.0017, while the proposed C-CSO based Deep LSTM obtained lower MSE of 0.0014 for 90% training data.

Table 1: Comparative Discussion

Metrics/Methods	Deep neural network	Spectral mixture analysis	Deep LSTM	Proposed C-CSO based Deep LSTM
MSE	0.7797	0.8882	0.0017	**0.0014**
Accuracy	54.616	67.315	80.553	**81.200**

5. Conclusion

In this research, an effective method named C-CSO based Deep LSTM is proposed to perform the change detection in the forest area. Initially, two satellites images at the same location with different time instances are selected to perform the change prediction process. The input images are pre-processed to eliminate the noise and artifacts present in the images in order to enhance further processing. The pre-processed results are allowed to the vegetation module, where the features, like Ratio Vegetation index, Enhanced Vegetation Index, Green Chlorophyll Index, and Green Leaf Area Index are extracted separately from two images. The pixel value location of each image along with the extracted features is used to perform the segmentation process using the active contour model. The segmented results of the two images are mapped to find the change detection and with the results of change detection, the change prediction is done using the proposed C-CSO based Deep LSTM. The performance attained by the proposed classifier is revealed using the metrics, like MSE and accuracy with the values of 0.0014 and 81.200%, respectively. In future, the performance of change prediction in the forest area is increased using some other optimization algorithm.

Compliance with Ethical Standards

Conflicts of interest: Authors declared that they have no conflict of interest.

Human participants: The conducted research follows ethical standards and the authors ensured that they have not conducted any studies with human participants or animals.

References

[1] Khan, S.H., He, X., Porikli, F. and Bennamoun, M., "Forest change detection in incomplete satellite images with deep neural networks," IEEE Transactions on Geoscience and Remote Sensing, vol.55, no.9, pp.5407-5423, 2017.

[2] Karimi, N., Golian, S. and Karimi, D., "Monitoring deforestation in Iran, Jangal-Abr Forest using multi-temporal satellite images and spectral mixture analysis method," Arabian Journal of Geosciences, vol.9, no.3, pp.214, 2016.

[3] Chance, C.M., Hermosilla, T., Coops, N.C., Wulder, M.A. and White, J.C., "Effect of topographic correction on forest change detection using spectral trend analysis of Landsat pixel-based composites," International journal of applied earth observation and geoinformation, vol.44, pp.186-194, 2016.

[4] Stylianidis, E., Akca, D., Poli, D., Hofer, M., Gruen, A., Sanchez Martin, V., Smagas, K., Walli, A., Altan, O., Jimeno, E. and Garcia, A., "FORSAT: a 3D forest monitoring system for cover mapping and volumetric 3D change detection," International Journal of Digital Earth, pp.1-32, 2019.

[5] Karila, K., Yu, X., Vastaranta, M., Karjalainen, M., Puttonen, E. and Hyyppä, J., "TanDEM-X digital surface models in boreal forest above-ground biomass change detection," ISPRS Journal of Photogrammetry and Remote Sensing, vol.148, pp.174-183, 2019.

[6] Vázquez-Quintero, G., Solís-Moreno, R., Pompa-García, M., Villarreal-Guerrero, F., Pinedo-Alvarez, C. and Pinedo-Alvarez, A, "Detection and projection of forest changes by using the Markov Chain Model and cellular automata," Sustainability, vol.8, no.3, pp.236, 2016.

[7] Healey, S.P., Cohen, W.B., Yang, Z., Brewer, C.K., Brooks, E.B., Gorelick, N., Hernandez, A.J., Huang, C., Hughes, M.J., Kennedy, R.E. and Loveland, T.R., "Mapping forest change using stacked generalization: An ensemble approach," Remote Sensing of Environment, vol.204, pp.717-728, 2018.

[8] Marinelli, D., Paris, C. and Bruzzone, L, "A novel approach to 3-d change detection in multitemporal lidar data acquired in forest areas," IEEE Transactions on Geoscience and Remote Sensing, vol.56, no.6, pp.3030-3046, 2018.

[9].Gebrehiwot S, Bewket W, Gärdenäs A, Bishop K, "Forest cover change over four decades in the Blue Nile Basin Ethiopia: comparison of three watersheds," Reg Environ Change, vol.14, pp.253–266, 2014.

[10]Avtar R, Sawada H, "Use of DEM data to monitor height changes due to deforestation," Arab J Geosci, vol.6, pp.4859–4871, 2013.

[11]Lu, D., Mausel, P., Brondizio, E. and Moran, E., "Change detection techniques," International journal of remote sensing, vol.25, no.12, pp.2365-2401, 2004.

[12]Coppin, P.R. and Bauer, M.E., "Digital change detection in forest ecosystems with remote sensing imagery," Remote sensing reviews, vol.13, no.4, pp.207-234, 1996.

[13]Collins, J.B. and Woodcock, C.E., "An assessment of several linear change detection techniques for mapping forest mortality using multitemporal Landsat TM data," Remote sensing of environment, vol.56, no.1, pp.66-77, 1996.

[14] Desclée, B., Bogaert, P. and Defourny, P., "Forest change detection by statistical object-based method," Remote Sensing of Environment, vol.102, pp.1-2, pp.1-11, 2006.

[15] C. Huang *et al.*, "Use of a dark object concept and support vector machines to automate forest cover change analysis," Remote Sens. Environ., vol. 112, no. 3, pp. 970–985, 2008.

[16]D. Tomowski, M. Ehlers, and S. Klonus, "Colour and texture based change detection for urban disaster analysis," in Proc. Joint Urban Remote Sens. Event (JURSE), pp. 329–332, 2011.

[17] M.-L. Nordberg and J. Evertson, "Vegetation index differencing and linear regression for change detection in a Swedish mountain range using Landsat TM and ETM+ imagery," Land Degradation Develop., vol. 16, no. 2, pp. 139–149, 2005.

[18] C. Huang, S. N. Goward, J. G. Masek, N. Thomas, Z. Zhu, and J. E. Vogelmann, "An automated approach for reconstructing recent forest disturbance history using dense landsat time series stacks," Remote Sens. Environ., vol. 114, no. 1, pp. 183–198, 2010.

[19] K. Nackaerts, K. Vaesen, B. Muys, and P. Coppin, "Comparative performance of a modified change vector analysis in forest change detection," Int. J. Remote Sens., vol. 26, no. 5, pp. 839–852, 2005.

[20] A. Versluis and J. Rogan, "Mapping land-cover change in a haitian watershed using a combined spectral mixture analysis and classification tree procedure," Geocarto Int., vol. 25, no. 2, pp. 85–103, 2010.

[21] B. C. Pijanowski, S. Pithadia, B. A. Shellito, and K. Alexandridis, "Calibrating a neural network-based urban change model for two metropolitan areas of the Upper Midwest of the United States," Int. J. Geograph. Inf. Sci., vol. 19, no. 2, pp. 197–215, 2005.

[22] J. Im and J. R. Jensen, "A change detection model based on neighborhood correlation image analysis and decision tree classification," Remote Sens. Environ., vol. 99, no. 3, pp. 326–340, 2005.

[23]Sharif, M., Hussain, A., Jaffar, M.A. and Choi, T.S., "Fuzzy similarity based non local means filter for Rician noise removal," Multimedia tools and applications, vol.74, no.15, pp.5533-5556, 2015.

[24]Xingjian, S.H.I., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K. and Woo, W.C., "Convolutional LSTM network: A machine learning approach for precipitation now casting," In Advances in neural information processing systems, pp. 802-810, 2015.

[25] Global PALSAR-2/PALSAR/JERS-1 Mosaic and Forest / Non-forest Map, "http://www.eorc.jaxa.jp/ALOS/en/palsar_fnf/data/index.htm", accessed on May 2019.

[26]Meng, X., Liu, Y., Gao, X. and Zhang, H., "A new bio-inspired algorithm: chicken swarm optimization," In proceedings of International conference in swarm intelligence, pp.86-94, 2014.

[27]Askarzadeh, A., "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm," Computers & Structures, vol.169, pp.1-12, 2016.

[28] Xingjian, S.H.I., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K. and Woo, W.C., "Convolutional LSTM network: A machine learning approach for precipitation nowcasting", In Advances in neural information processing systems, pp. 802-810, 2015.

[29] Yin, H., Weng, L., Li, Y., Xia, M., Hu, K., Lin, H. and Qian, M., "Attention-guided siamese networks for change detection in high resolution remote sensing images", International Journal of Applied Earth Observation and Geoinformation, Vol. 117, pp.103206, 2023.

[30] Aniah, P., Bawakyillenuo, S., Codjoe, S.N.A. and Dzanku, F.M., "Land use and land cover change detection and prediction based on CA-Markov chain in the savannah ecological zone of Ghana", Environmental Challenges, Vol.10, pp.100664, 2023.

[31] Pokhariya, H.S., Singh, D.P. and Prakash, R., "Land use/land cover change detection and forecasting using GEE and hybrid Markov-CA model in the Nainital, a district of Uttarakhand State, India", 2023.

[32] Lavanya, K., Mahendran, A., Selvanambi, R., Mazzara, M. and Hemanth, J.D., "Tunicate Swarm Algorithm with Deep Learning Based Land Use and Cover Change Detection in Nallamalla Forest India", Applied Sciences, Vol. 13(2), pp.1173, 2023.

[33] Ahmad, M.N., Shao, Z., Javed, A., Islam, F., Ahmad, H.H. and Aslam, R.W., "The Cellular Automata Approach in Dynamic Modelling of Land Use Change Detection and Future Simulations Based on Remote Sensing Data in Lahore Pakistan", Photogrammetric Engineering & Remote Sensing, Vol. 89(1), pp.47-55, 2023.

[34] Xiang, J., Xing, Y., Wei, W., Yan, E., Jiang, J. and Mo, D., "Dynamic Detection of Forest Change in Hunan Province Based on Sentinel-2 Images and Deep Learning", Remote Sensing, Vol. 15(3), pp.628, 2023.

[35] Affonso, A.A., Mandai, S.S., Portella, T.P., Quintanilha, J.A., Conti, L.A. and Grohmann, C.H., "A Comparison between Supervised Classification Methods: Study Case on Land Cover Change Detection Caused by a Hydroelectric Complex Installation in the Brazilian Amazon", Sustainability, Vol. 15(2), pp.1309, 2023.

[36] Shimizu, K., Murakami, W., Furuichi, T. and Estoque, R.C., "Mapping Land Use/Land Cover Changes and Forest Disturbances in Vietnam Using a Landsat Temporal Segmentation Algorithm", Remote Sensing, Vol. 15(3), pp.851, 2023.