Measuring Moran's I in a Cost-Efficient Manner to Describe a Land-Cover Change Pattern in Large-Scale Remote Sensing Imagery

Monidipa Das, *Student Member, IEEE*, and Soumya K. Ghosh, *Member, IEEE*

***Abstract*—Detection and analysis of a land-cover change pattern from remotely sensed imagery have gained increasing research interests in recent years. A number of spatial statistics and landscape pattern metrics have been explored for this purpose. Moran's index (Moran's I) of spatial autocorrelation is one such spatiostatistical measure, which has been proved to be useful in characterizing the land-cover change, especially in Landsat data. However, since the Moran's I estimation needs to deal with spatial weight between each pair of spatial data objects, it becomes almost unfeasible to apply Moran's I in the case of large-scale remote sensing data, containing several millions of pixels. This paper proposes a method for computing Moran's I in the Hadoop MapReduce framework and thereby helps in describing spatial patterns in large-scale remotely sensed data. The contributions of the work include: 1) the exhaustive description of the Mapper and Reducer implementation for cost-effective estimation of Moran's I, and 2) the computational complexity analysis of the respective algorithms. Furthermore, two case studies have been presented, considering both the rook case and the queen case of spatial contiguity. Case Study 1 demonstrates the computational efficiency of the proposed implementation, and Case Study 2 illustrates an application of Moran's I in describing the urban sprawling pattern in two large spatial zones in Kolkata, India.**

***Index Terms*—Land-cover change pattern, large-scale data, Moran's index (Moran's I), MapReduce, remote sensing imagery.**

I. Introduction

THE change in land use/land cover is a key driver of the global change, and it can have a significant effect on the socioeconomical, ecological, and other environmental systems, like climate [1], [2]. Therefore, research on detection and analysis of a land-cover change pattern has acquired increasing interest in present days, and in this regard, the satellite remote sensing plays a crucial role by providing an important source of land-use/land-cover data [3], [4]. Variants of spatial statistics and landscape pattern metrics have been explored for describing land-cover dynamics in remotely sensed data. Among these techniques/measures, the Moran's index (Moran's I) of spatial autocorrelation is proved to provide a comparatively rapid and automated technique [5] for the identification of interesting spatial patterns, such as dispersion, randomness, clustering, etc.

Several land-cover change processes, such as deforestation, urban sprawl, etc., are manifested in small spatial scale and, thus, need high-resolution remote sensing data for better analysis [6]. However, since the Moran's I estimation deals with spatial weight between each pair of spatial data objects, it becomes almost unfeasible to conventionally measure Moran's I from such large-scale raster data, containing several millions of pixels. In this work, we have used a cost-efficient method for measuring Moran's I to facilitate the process of characterizing the spatial change pattern in large-scale remotely sensed imagery. This work can be treated as a more detailed presentation of our work proposed in [7]. In the present paper, the proposed approach has been described along with additional datasets and comparisons in an enhanced experimental section.

A. Related Work

Down through the years, Moran's I has been adopted in several research works for characterizing the spatial pattern in land-cover data [8]. For example, Overmars *et al.* [9] have proposed a mixed regressive-spatial autoregressive model, which uses Moran's I of spatial autocorrelation for spatial analysis purpose. The model has been employed in analyzing the land-use data in Ecuador. Estiri [10] has used Moran's I for quantitatively identifying the urban sprawl from remotely sensed data. The study also found it useful to develop a regression model to predict the urban sprawl based on the outcomes of Moran's I analysis on land-cover data. In [11], Pierre *et al.* have utilized Moran's I as global statistical metrics to analyze landscape disturbances. The 30-m resolution raster National Land Cover Dataset has been used for this purpose. Read and Lam, in [5], have analyzed a number of spatial methods for the detection of a land-cover change and the characterization of a land-cover pattern. The study demonstrates Moran's I to be more useful than standard landscape indices for characterizing the spatial pattern, especially in Landsat-TM data. Chen *et al.* [12] have used Moran's I of spatial autocorrelation to describe the spatial pattern of change in lake area and also to identify the clusters of lakes with similar change trends. In the work of Roberts *et al.* [13], Moran's I has been proved to be useful for analyzing forest fragmentation as well. An exhaustive study on application of Moran's I in urban growth analysis has been performed by Tsai

Manuscript received September 15, 2016; revised December 21, 2016; accepted January 9, 2017. *(Corresponding author: Monidipa Das.)*

The authors are with the Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur 721302, India (e-mail: monidipadas@hotmail.com; skg@iitkgp.ac.in).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTARS.2017.2660766

[14]. The study has been conducted with vector data, and it concludes that the Moran's coefficients are low, intermediate, and high for decentralized, polycentric, and monocentric sprawling patterns, respectively.

However, using Moran's I to characterize the landscape pattern in remotely sensed imagery is a less explored area [5]. A high computational cost of measuring Moran's I in large-scale raster data is a crucial issue in behind [15]. The same reason also limits the existing techniques to be applied on high-resolution remote sensing imagery.

B. Challenges in Computing Moran's I from Large-Scale Remotely Sensed Data

Moran's I, developed by P. A. P. Moran in 1950 [16], [17], is a widely used measure of spatial autocorrelation. Numerically, Moran's I (I) can be expressed as follows:

$$I = \frac{n \times (\sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}.(x_i - \bar{x})(x_j - \bar{x}))}{(\sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}) \times (\sum_{i=1}^{n} (x_i - \bar{x})^2)} \quad (1)$$

where x_i and x_j are observed values of a spatial feature at locations i and j, respectively. $\bar{x}$ is the mean of observed values in all the sites, n is the number of observation locations/sites, and w_{ij} is the weight, defined based on the spatial proximity between locations i and j.

The values of I range between -1 and +1. Negative values indicate negative spatial autocorrelations and positive values indicate positive spatial autocorrelations. A zero value indicates existence of a random spatial pattern.

However, since the Moran's I computation needs to deal with weight w_{ij} between each pair of spatial locations $< i, j >$, it becomes computationally intensive in the case where the total number of observation locations or sites (n) becomes very large. A critical situation arises while estimating Moran's I in a very large raster data, like satellite remote sensing image, containing millions of pixels or spatial grids, each of which represents an observation location. One such example scenario is depicted in Fig. 1. It can be visualized from the figure (refer to Fig. 1) that the computation time for Moran's I is increasing in polynomial order with respect to the total pixels under study. Moreover, from the trend line equation, it can be inferred that, under similar experimental setup, it may take years to estimate Moran's I in a satellite image having 10 million pixels. Furthermore, storing the large weight matrix in a single stand-alone computer system also becomes difficult during Moran's I computation. In these circumstances, a cost-effective solution for measuring Moran's I in large-scale raster data is severely needed.

C. Contributions

The present work provides a cost-efficient solution for characterizing a spatial pattern in large-scale raster data. This has been achieved by proposing a MapReduce [18] implementation of Moran's I estimation, which drastically reduces the computation time. The MapReduce framework is inherently capable of efficiently processing huge volume of data in a parallel and distributed fashion. However, to the best of our knowledge, this is the first attempt in using the MapReduce implementation of Moran's I for characterizing the land-cover pattern in large-scale remote sensing imagery. The major contributions in this work are summarized as follows:

1) proposing a cost-efficient solution of describing a land-cover pattern from large-scale satellite imagery based on the MapReduce implementation of Moran's I;
2) extensively illustrating the map and reduce algorithms to measure Moran's I in large-scale remotely sensed data;
3) theoretically analyzing the computational complexity of the proposed MapReduce implementation of Moran's I;
4) empirically analyzing the runtime performance of the proposed algorithm in comparison with the single-machine setup;
5) applying the proposed cost-efficient measure of Moran's I to describe the pattern of the urban sprawl in two large spatial zones in Kolkata, India.

The rest of this paper is organized as follows. The proposed approach of characterizing a land-cover change pattern using a cost-efficient measure of Moran's I has been illustrated in Section II. The theoretical performance analysis of the proposed implementation of Moran's I has been presented in Section III. The experimental results have been thoroughly discussed in Section IV with respect to two case studies, and finally, the concluding remarks have been made in Section V.

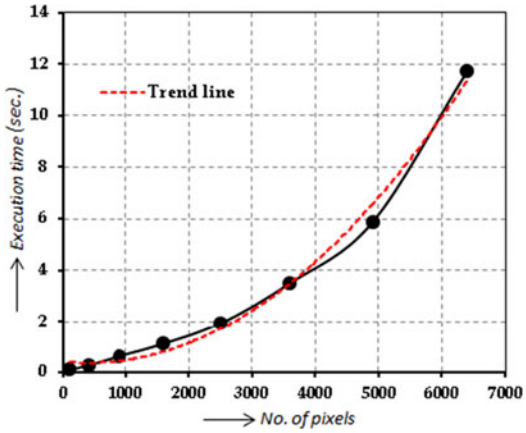

Fig. 1. Change in the computation time of Moran's I versus change in the total pixel count in the raster data.[1]

II. Proposed Approach: Describing a Land-Cover Change Pattern Using Cost-Efficient Moran's I

This section provides a detailed description of the proposed approach for describing a land-cover change pattern by using a cost-efficient measure of Moran's I. The central concentration of the section has been kept on explaining the MapReduce implementation of Moran's I.

As shown in Fig. 2, the overall approach consists of a data preprocessing step followed by Moran's I computation in the MapReduce framework and land-cover pattern analysis. In order to describe the overall process, let us assume that each of the

[1]Package: *R-Tool* (System setup: 64-bit OS and 4-GB RAM).

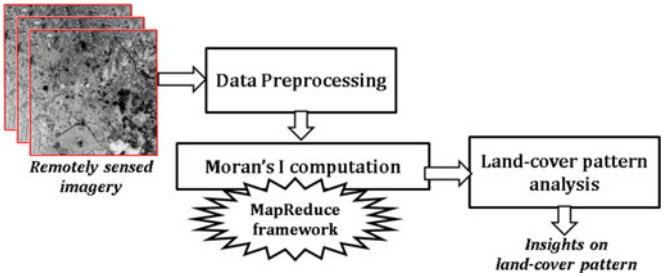

Fig. 2. Proposed approach for characterizing a land-cover change pattern using cost-efficient Moran's I.

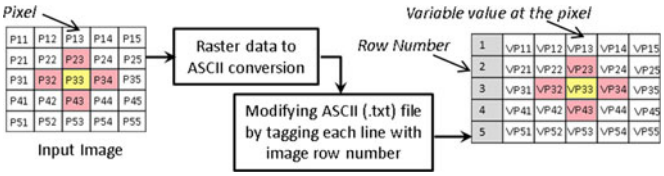

Fig. 3. Illustration of the step of data preprocessing.

input imagery R contains r number of rows and c number of columns, i.e., a total of $n = (r \times c)$ number of pixels.

A. Data Preprocessing

In this step, the remote sensing raster data are first converted into ASCII format and further processed by appending the row number at the beginning of each row. The idea has been explained through Fig. 3. The figure illustrates the preprocessing of a 5×5 image, where a pixel at the ith row/line and the jth column has been denoted with P_{ij}, and the variable value corresponding to the pixel has been denoted by VP_{ij}. Since the MapReduce execution framework automatically splits the input data records into physical blocks and distributes them among different mappers, the original position/location and, hence, the spatial relationships among the pixels can be lost. Inclusion of this row number information can help to track the original pixel location in the input file. The processed ASCII file is then fed to the next step for Moran's I computation.

B. Moran's I Computation in the MapReduce Framework

Given a preprocessed remote sensing image (raster data), this step computes Moran's I in a cost-efficient manner, based on the MapReduce framework. Typically, a MapReduce process consists of three stages: map, shuffle, and reduce. The work only illustrates the map and reduce function for calculating Moran's I from the input raster. The shuffle step is automatically performed using the built-in logic of the MapReduce programming model.

1) Mapper Implementation: The proposed map function is presented through Algorithm 1. In the map stage, the mapper takes the input $<$ key; value $>$ pair in following format:

$$< \text{line offset; line/row content} >$$

A typical $value$ corresponding to the ith line/row looks as follows:

$$(i, VP_{i1}, \ldots, VP_{ic})$$

Algorithm 1: Map_function(key,$value$).

Input : key=line offset;
$value$=line (row) content:$(i, VP_{i1}, VP_{i2}, \cdots, VP_{ic})$, where VP_{ij} is the value of pixel P_{ij} in j-th column of the row.
Output: $< I_{key}, I_{val} >$: Intermediate $< key, value >$ pair.

LP_{ij} = location of P_{ij} in the original raster image;
$LP_{ij}^{N_k}$ = location of the k-th neighbor N_k of P_{ij} $(k = 1, \cdots, b)$;
b = number of neighbors for P_{ij} as per the given spatial contiguity;
m = mean of all the feature values in the input data;

for *each* $VP_{ij} \in value$ **do**
  $I_{key} = LP_{ij}$; $I_{val} = (VP_{ij}, LP_{ij}, 1, (VP_{ij} - m))$;
  emit(I_{key},I_{val});
  for *each neighbor* N_k $(k = 1, \cdots, b)$ *of* P_{ij} **do**
    $I_{key} = LP_{ij}^{N_k}$; $I_{val} = (VP_{ij}, LP_{ij}, 1, (VP_{ij} - m))$;
    emit(I_{key},I_{val});
  end
end

where VP_{ij} is the value of a spatial feature at pixel P_{ij} in the jth column of the row.

Now, let, for any pixel P_{ij}, the number of neighbors is b. Then, as per the proposed map function, for each row element (except the first field representing the row number), $(b+1)$ number of intermediate $<$ key; value $>$ pairs are generated in the following manner. For each VP_{ij} in the line, the generated set of intermediate $<$ key; value $>$ pairs becomes $\{< LP_{ij};\ (VP_{ij}, LP_{ij}, 1, (VP_{ij} - m)) >,\ < LP_{ij}^{N_1};\ (VP_{ij}, LP_{ij}, 1, (VP_{ij} - m)) >,\ \cdots,\ < LP_{ij}^{N_b};\ (VP_{ij}, LP_{ij}, 1,\ (VP_{ij} - m)) >\}$, where $LP_{ij}^{N_k}$ is the location of the kth neighbor N_k of P_{ij}, LP_{ij} is the location of P_{ij}, VP_{ij} is the value of spatial feature at P_{ij}, and m is the mean of all the feature values in the input raster. The maximum value of b in the rook case of spatial contiguity is 4, and that in the queen case is 8. The $LP_{ij}^{N_k}$ $(k = 1, \ldots, b)$ values can be determined in terms of i and j.

An example scenario has been illustrated in Fig. 4 with respect to pixel P_{33} and the rook case of spatial contiguity. As per the rook case of contiguity, the neighbors of pixel P_{33} are P_{23}, P_{32}, P_{34}, and P_{43}, respectively, as highlighted with shades/patches. Now, let us consider the left-most mapper, which has been assigned the third row as split data. Therefore, as per the proposed approach, the initial $<$ key; value $>$ pair for this mapper becomes

$$< 3; VP_{31}, VP_{32}, VP_{33}, VP_{34}, VP_{35} >$$

Then, while processing the element VP_{33}, $(4+1) = 5$ number of intermediate $<$ key; value $>$ pairs will be produced as follows:

$$< LP_{23}; (VP_{33}, LP_{33}, 1, (VP_{33} - m)) >$$
$$< LP_{32}; (VP_{33}, LP_{33}, 1, (VP_{33} - m)) >$$
$$< LP_{33}; (VP_{33}, LP_{33}, 1, (VP_{33} - m)) >$$
$$< LP_{34}; (VP_{33}, LP_{33}, 1, (VP_{33} - m)) >$$
$$< LP_{43}; (VP_{33}, LP_{33}, 1, (VP_{33} - m)) >$$

where $m = \frac{1}{25} \sum_{i=1}^{5} \sum_{j=1}^{5} VP_{ij}$.

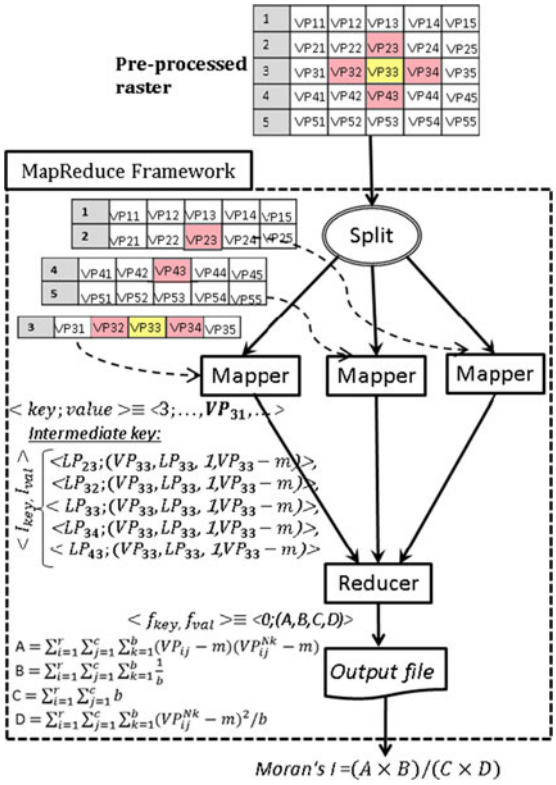

Fig. 4. Illustrating example for measuring Moran's I in the MapReduce framework.

2) Reducer Implementation: The mapper function has been designed in such a way that each of the values associated with a single key in the reduce stage are nothing but the details of each of its neighbors including itself. The set of values for a single key LP_{ij} becomes $\{(VP_{ij},\ LP_{ij},\ 1,\ (VP_{ij} - m)), (VP_{ij}^{N_1}, LP_{ij}^{N_1},\ 1,\ (VP_{ij}^{N_1} - m)), \ldots,\ (VP_{ij}^{N_b}, LP_{ij}^{N_b},\ 1, (VP_{ij}^{N_b} - m))\}$, where $LP_{ij}^{N_k}$ is the location of the kth neighbor N_k of P_{ij}, LP_{ij} is the location of P_{ij}, VP_{ij} is the spatial feature value at P_{ij}, $VP_{ij}^{N_k}$ is the spatial feature value at the kth neighbor of P_{ij}, m is the mean of all the feature values in the input data, and b is the number of neighbors for the pixel P_{ij}.

With this neighborhood information, output $<$ key; value $>$ pair for each key is at first generated as follows: $< 0; (\sum_{k=1}^{b}(VP_{ij} - m).(VP_{ij}^{N_k} - m),\ \sum_{k=1}^{b}\frac{1}{b},\ b,\ \sum_{k=1}^{b}(VP_{ij}^{N_k} - m)^2/b) >$. Then, each of these values are further summed over the new $key = 0$ to generate the final output $<$ key; value $>$ pair:

$$< 0; (A, B, C, D) >$$

where

$$A = \sum_{i=1}^{r}\sum_{j=1}^{c}\sum_{k=1}^{b}(VP_{ij} - m).(VP_{ij}^{N_k} - m) \tag{2}$$

$$B = \sum_{i=1}^{r}\sum_{j=1}^{c}\sum_{k=1}^{b}\frac{1}{b} = nb/b = n \tag{3}$$

Algorithm 2: Reduce_function(key,$value$).

Input : key=location LP_{ij} of a pixel P_{ij} in j-th column of the i-th row in original raster; $value$=A set of values of the form (v_1, v_2, v_3, v_4). $v_1 \in \left\{VP_{ij} \cup VP_{ij}^{N_k}\right\}$,$v_2 \in \left\{LP_{ij} \cup LP_{ij}^{N_k}\right\}$,$v_3 = 1$, and $v_4 \in \left\{(VP_{ij} - m) \cup (VP_{ij}^{N_k} - m))\right\}$, where $k \in \{1, \cdots b\}$

Output: $< f_{key}, f_{val} >$: Final $< key, value >$ pair.

```
m= mean of all the feature values in the input data;
LP_ij= location of P_ij in the original raster image;
VP_ij= spatial feature value at P_ij;
LP_ij^{N_k} = location of the k-th neighbor N_k of P_ij;
b= number of neighbors for P_ij as per the given spatial contiguity;
A = 0; B = 0; C = 0; D = 0; /* initialization */
if key ≠ 0 then
    for each val ∈ value do
        if key ≠ v_2 then
            A = A + (VP_ij − m)(v_1 − m); B = B + (1/b);
            C = C + 1; D = D + v_4^2/b;
        end
    end
    emit(0,(A,B,C,D));
end
else
    for each val ∈ value do
        A = A + v_1; B = B + v_2; C = C + v_3; D = D + v_4;
    end
    f_key = key; f_val = (A, B, C, D);
    emit(f_key, f_val);
end
```

$$C = \sum_{i=1}^{r}\sum_{j=1}^{c} b \tag{4}$$

$$D = \sum_{i=1}^{r}\sum_{j=1}^{c}\sum_{k=1}^{b}(VP_{ij}^{N_k} - m)^2/b. \tag{5}$$

Now, the Moran's I value can be estimated based on these A, B, C, D, as stored in the output file. Moran's I, thus, becomes

$$I = (A \times B)/(C \times D). \tag{6}$$

As per this design, the proposed reduce function is applicable for a single reduce task setup. The detailed structure for the proposed reduce function is presented in Algorithm 2.

C. Land-Cover Pattern Analysis

This step takes as input the estimated Moran's I and generates various insights on the land-cover pattern. The values of Moran's I along with a statistical significance (Z-score) indicates a cluster pattern in the distribution of spatial features, in the case where the value is positive; dispersed pattern, in the case where the value is negative; and randomness, in the case where the value is zero. Moreover, Moran's I can also be utilized to distinguish between compactness and sprawl at both metropolitan and local levels. For example, a high value of Moran's I indicates monocentric sprawling [14], an intermediate value of the Moran coefficient indicates polycentric sprawling, and a low value of the Moran coefficient indicates decentralized sprawling pattern at the metropolitan level.

III. Analyzing Computational Complexity of MapReduce Implementation of Moran's I

This section analyzes the computational complexity of measuring Moran's I using the proposed MapReduce implementation. Let us assume that the total number of rows and columns

in an input image is r and c, respectively. Therefore, each image raster consists of $n = (r \times c)$ number of pixels. Also consider that the number of neighbors for any pixel is b.

Now, as per the proposed mapper implementation, for a single $<$ key; value $>$ pair, the number of generated intermediate $<$ key; value $>$ pair is $(c \times (b+1))$. Therefore, the maximum space required to store all the intermediate keys for the entire input data is

$$S_{\max} = O(rc(b+1)) = O(n(b+1)). \tag{7}$$

However, for large-scale raster data, $b << n$. Therefore, (7) yields the following:

$$S_{\max} = O(n(b+1)) << O(n^2). \tag{8}$$

Moreover, since the total number of generated intermediate $<$ key; value $>$ pair is $(r \times c \times (b+1))$, the reducer takes maximum $O(rc(b+1)) = O(n(b+1))$ time for computation. However, as mentioned previously, for large-scale raster data, $b << n$. Therefore, the maximum time taken by the reducer is

$$T_{\max} = O(n(b+1)) << O(n^2). \tag{9}$$

Thus, the problem of handling an $n \times n$ spatial weight matrix is resolved in the proposed MapReduce implementation of Moran's I.

IV. Experimentation

The experimentation has been carried out on a Hadoop cluster running Hadoop version 2.3.0. Each of the 25 nodes in the cluster has two Intel(R) Xeon(R) E5-2630 v2 (2.60 GHz) CPUs, six cores per CPU, 132-GB RAM, and 256.6-TB hard disk. Two different case studies have been considered for the present purpose. Case Study 1 has been used to demonstrate the computational efficiency of the proposed implementation in comparison with single stand-alone machine setup, whereas Case Study 2 has been performed to illustrate the application of characterizing the urban sprawling pattern in large-scale remote sensing imagery by using the proposed cost-efficient measure of Moran's I.

A. Case Study 1: Empirically Analyzing the Computational Efficiency of the Proposed Implementation of Moran's I

1) Dataset: Case Study 1 has been carried out with the normalized difference vegetation index (NDVI) data from a set of six satellite imagery of varying dimensions, as depicted in Fig. 5(a)–(f). In Fig. 5(a)–(f), the NDVI datasets have been represented with gray-scale images, where the darkest shade (black) indicates the lowest value of NDVI and the lightest shade (white) indicates the highest value of NDVI. The original FCC (false color composite: near infrared (red), red (green), green (blue))[2] images have also been placed side by side for better understanding. The primary source of these raster data is the Landsat-7 ETM+ satellite imagery from the Land Process Distributed Active Archive Center of the United States Geological Survey (USGS) [19]. Later, ERDAS IMAGINE tool has been utilized to generate the NDVI raster [20] from the input raw satellite imagery.

[2]http://earthobservatory.nasa.gov/Features/FalseColor/page6.php

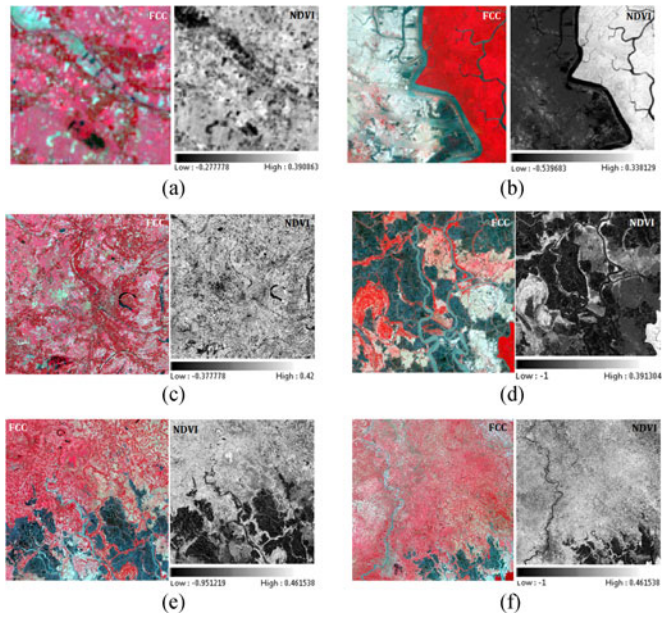

Fig. 5. Raster datasets for Case Study 1. (a) Dataset 1. (b) Dataset 2. (c) Dataset 3. (d) Dataset 4. (e) Dataset 5. (f) Dataset 6.

TABLE I
Results of the Proposed MapReduce Implementation of Moran's I Considering First-Order Spatial Contiguity

Dataset	Pixel Count	Estimated Moran's I	
		Queen case	Rook case
Dataset 1	1.0×10^4	0.737	0.806
Dataset 2	4.0×10^4	0.972	0.972
Dataset 3	2.5×10^5	0.738	0.798
Dataset 4	1.0×10^6	0.950	0.964
Dataset 5	4.0×10^6	0.944	0.958
Dataset 6	2.5×10^7	0.920	0.942

The details of each raster image can be found in Table I.

2) Results and Discussion: For each experimental dataset, the estimated Moran's I considering the queen case and the rook case of spatial contiguity has been presented in Table I. The correctness of estimated values of Moran's I for smaller datasets have been validated with respect to the Moran's I measure from spatial R [21] and ArcGIS Tool [22]. The amount of CPU time spent (considering four map tasks) for each dataset has been depicted in Fig. 6. Moreover, the performance of the proposed MapReduce algorithm has been studied with respect to the change in the number of mapper tasks involved (refer to Fig. 7). From the figures, following inferences can be drawn.

1) Fig. 6 shows that the CPU time spent increases linearly with the number of pixels. However, as per the trend line equation, the increment is slow for very large scale datasets, and negligible (almost constant) otherwise.
2) It can be noted from Fig. 7 that, initially, the computation time for the proposed Moran's I implementation

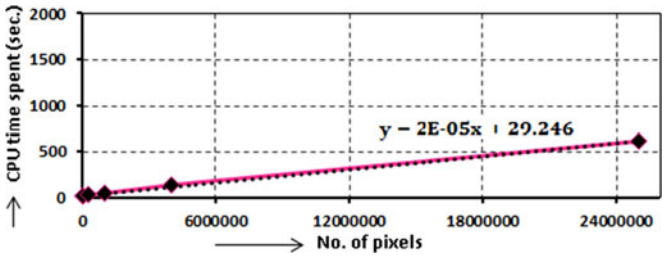

Fig. 6. Change in the CPU time spent with respect to the increment in the pixel count.

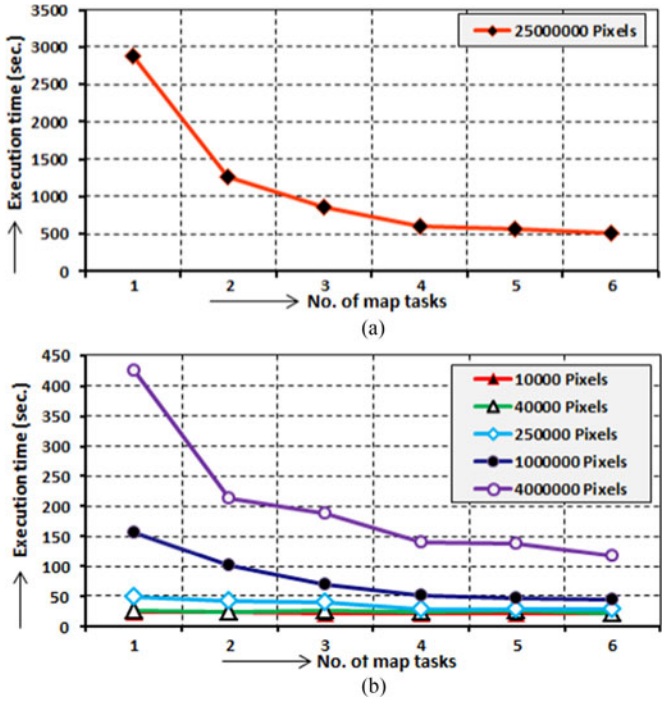

Fig. 7. Change in the execution time with respect to the increment in the number of map tasks.

drastically decreases with the increment in the number of map tasks. However, the decrement is negligible beyond four map tasks in each case. So, in an optimum setup, the proposed algorithm can be executed using four map tasks.

3) Since a single-machine setup can work at most as efficiently as a single-map task setup, from Fig. 7, it can also be inferred that for a very large dataset, the proposed MapReduce algorithm is able to reduce average 78% of the computation time in the single-machine setup.

4) The trend of change in computation time requirement, as derived from Fig. 7, is depicted in Fig. 8. The trend equations in Fig. 8 reveal that a single-machine setup (equivalent to a single-map task) can take days to compute Moran's I from just a 30 km $\times$ 30 km, 1-m resolution satellite imagery. However, our proposed approach with four mapper tasks will take few hours to achieve the same. That is, the proposed approach is very much effective for high/very high resolution imagery, and the improvement is not always merely in few seconds. The effectiveness is more evident as the size of data (number of pixels) increases.

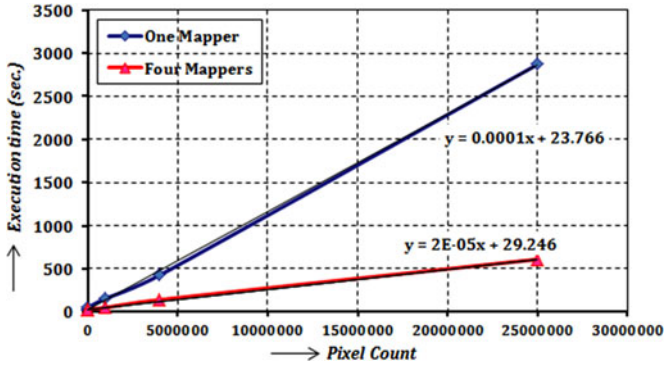

Fig. 8. Tendency of increasing execution time in Moran's I calculation.

B. Case Study 2: Characterizing the Urban Sprawl Pattern by Analyzing Large-Scale Remote Sensing Imagery

1) Dataset and Study Area: Case Study 2 has been performed to characterize the urban sprawling pattern in two large spatial zones in and around the city of Kolkata, India (refer to Fig. 9). The experimentation has been carried out with the normalized difference built-up index (NDBI) time-series raster data, captured at a gap of four years during the period 2001–2013. The primary source of these raster data is the Landsat-7 ETM+ satellite imagery from USGS [19]. Later, the ERDAS IMAGINE tool[3] has been used to generate the NDBI raster from the raw satellite imagery using the following equation:

$$\text{NDBI} = \frac{(\text{SWIR} - \text{NIR})}{(\text{SWIR} + \text{NIR})} \tag{10}$$

where SWIR and NIR are the top-of-the-atmosphere reflectance measurements, captured in the short-wave infrared (Band 5) and near-infrared (Band 4) spectral regions, respectively. Now, because of a high level of urbanization, study zone 1, which is near the Kolkata International Airport, has seen a continuous land-cover change in the overall region, especially during the period from 2005 to 2013. On the other side, study zone 2 is a quite larger zone, compared to zone 1. Therefore, the urban growth pattern in zone 2 is a bit scattered with respect to the whole region. The details of zones 1 and 2 have been summarized in Table II. The histogram plots of the distribution of NDBI in both the study zones have been depicted in Fig. 10. The plots have been generated with respect to discretized NDBI, which has been further normalized in the range: 0–255.

2) Results and Discussion: As per the proposed approach, the Moran's I values have been estimated considering the time series of NDBI imagery of the years 2001, 2005, 2009, and 2013 for both the study zones. The estimated Moran's I and the corresponding z-scores, with respect to the rook case and the queen case of spatial contiguity, have been presented in Tables III and IV, respectively, along with the details of computation time.

[3]http://www.hexagongeospatial.com/products/remote-sensing/erdas-imagine/overview

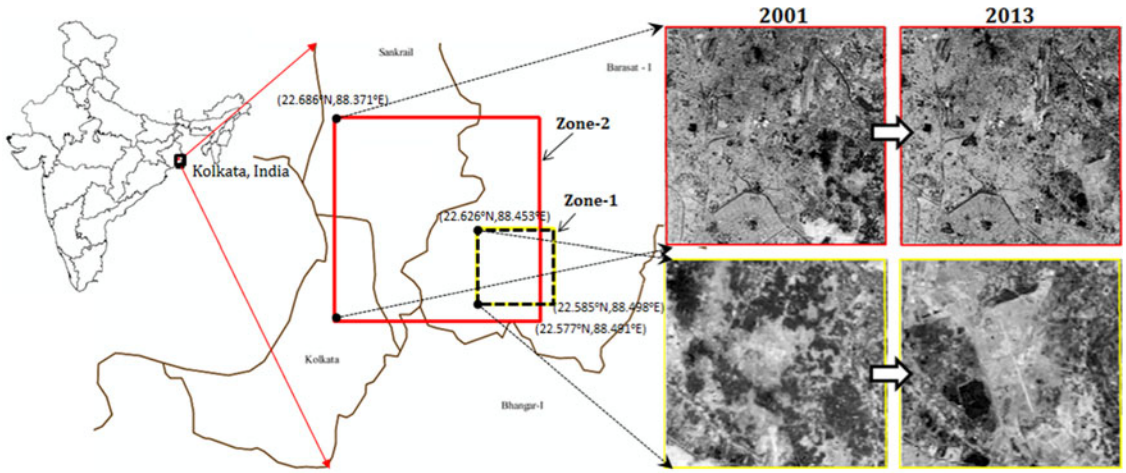

Fig. 9. Study area for Case Study 2: Kolkata, India.

TABLE II
DETAILS OF STUDY ZONES IN CASE STUDY 2

Zones	Bounding box details		
	No. of pixels	Top-Left	Bottom-Right
Zone 1	22 201	22.63°N, 88.45°E	22.59°N, 88.50°E
Zone 2	160 000	22.69°N, 88.37°E	22.58°N, 88.49°E

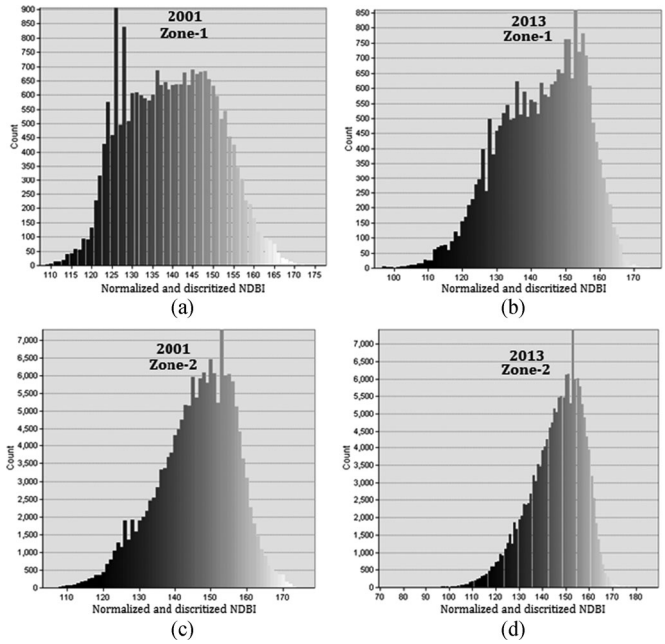

Fig. 10. Histogram plots of NDBI distribution (discretized and further normalized in the range 0–255) in Zone 1 and Zone 2.

In order to analyze the pattern of the urban sprawl in the study zone, the estimated Moran's indices have also been plotted in Fig. 11. On analyzing the tabulated values and the graphical plots, the following inferences can be drawn.

TABLE III
ESTIMATED MORAN'S I CONSIDERING THE ROOK CASE OF SPATIAL CONTIGUITY

		2001	2005	2009	2013	Time (in seconds)
Zone 1	Moran's I	0.8754	0.875	0.8726	0.8688	31 s
	Z-Score	259.95	259.83	259.12	257.99	
Zone 2	Moran's I	0.8516	0.8382	0.8482	0.8264	32 s
	Z-Score	680.41	669.71	677.70	660.28	

TABLE IV
ESTIMATED MORAN'S I CONSIDERING THE QUEEN CASE OF SPATIAL CONTIGUITY

		2001	2005	2009	2013	Time (in seconds)
Zone 1	Moran's I	0.8412	0.8410	0.836	0.8336	32 s
	Z-Score	352.72	352.64	350.54	349.54	
Zone 2	Moran's I	0.8105	0.7944	0.8069	0.7787	33 s
	Z-Score	915.21	897.03	911.14	879.30	

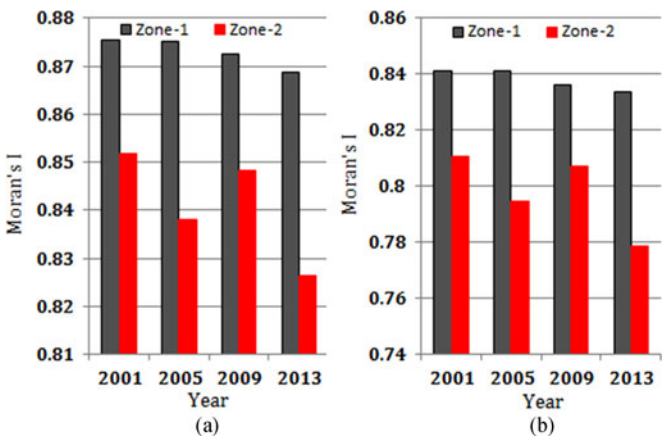

Fig. 11. Change in Moran's I during 2001–2013 in study Zone 1 and study Zone 2.

TABLE V
LAND-COVER CHANGE PATTERN AS GENERATED BY STATE-OF-THE-ART TECHNIQUES [8], [15]

Study Zones	State-of-the-art Techniques	2001	2005	2009	2013
Zone 1	Altieri *et al.* [15] (2014)	No sprawl	No sprawl	Partial Sprawl (with augmented size)	Partial Sprawl (with augmented size)
	Su *et al.* [8] (2011)	Several significant low–low clusters of PD	Several significant low–low clusters of PD	Significant high–high clusters of PD appears	Significant high–high clusters of PD increases
Zone 2	Altieri *et al.* [15] (2014)	No sprawl	Partial sprawl (with decreased size)	Partial sprawl (with augmented size)	Partial sprawl (with decreased size)
	Su *et al.* [8] (2011)	Several significant high–high clusters of PD	Several significant high–high clusters of PD	Several significant high–high clusters of PD	Several significant high–high clusters of PD

PD = Patch Density (A landscape metric; an indicator for landscape fragmentation)

1) It is evident from Tables III and IV that for both the study zones, the values of Moran's I are quite high with significantly large z-scores. This indicates a cluster pattern of a built-up index in both the zones. However, Fig. 11 shows a decreasing trend of Moran's I, especially in zone 1. The reason is that, initially (especially during 2001–2005), zone 1 was full of cultivable lands and water bodies. Therefore, a major part of the zone had a very low built-up index (NDBI) during this time period. Thus, the spatial distribution of NDBI produced a cluster pattern over low values of NDBI. The recent urbanization especially after 2005 has resulted in a rapid growth of the built-up area in this zone, which has affected the cluster pattern, and hence, the Moran's I values have started to decrease. This can also be validated from the histogram plots in Fig. 10. Similar insights can also be drawn for zone 2. However, since this zone is spread over a quite large region, the Moran's I values during 2001–2013 show a fluctuating behavior, indicating the scattered growth of the built-up area in this zone.
2) It can also be noted from Tables III and IV that though zone 2 has a significantly large number of pixels (160 000) compared to zone 1, the computation time for Moran's I in zone 2 is notably low, which demonstrates the effectiveness of the proposed approach in dealing with large-scale raster data.

In order to validate the urban sprawl pattern described by our proposed approach, a comparative study has been made with two state-of-the-art land-cover change analysis techniques, proposed in [15] and [8] respectively. Both the works [8] and [15] have utilized Moran's I of spatial autocorrelation for generating insights on the land-cover change.

In [15], Altieri *et al.* have proposed a scatterplot-based technique to identify and assess the urban sprawl. However, it is not free from information loss and, therefore, cannot regenerate the exact value of Moran's I. In our experimentation, the technique proposed in [15] has been implemented considering the binary data on an urban landscape pattern, generated using ArcGIS tool.[4] In the other work [8], Su *et al.* have used Moran's I statistics on various landscape metrics to characterize the degree of spatial dependence on landscape pattern changes over time. However, the method is not suitable for analyzing the land-cover change from large-scale remote sensing imagery. In our experimentation, the technique proposed in [8] has been implemented considering patch density of urban landscape, generated using FRAGSTATS software [23], for a 1.2 km $\times$ 1.2 km grid over the considered study zones.

The insights on the land-cover change pattern as generated by these state-of-the-art techniques have been summarized in Table V. The combined outcomes from both the techniques reveal that study zone 1 has encountered recent growth in the urban area, especially after 2005, whereas the major portions of study zone 2 belong to the urban area after 2001, and there also exists tendency of the landscape pattern change within small localities in zone 2. Thus, the comparative study shows that the outcomes of the state-of-the-art techniques support the results generated by our proposed approach.

[4]http://www.esri.com/software/arcgis/arcgis-for-desktop

V. CONCLUSION

The present work has proposed a cost-efficient approach for characterizing the land-cover change pattern from large-scale remotely sensed imagery. The approach is based on the MapReduce implementation of Moran's I, which can be effectively used in any other large-scale spatial data analysis as well. The proposed approach has been validated with two case studies using Landsat ETM+ satellite imagery. The experimental results demonstrate that the proposed approach is able to generate useful insights with 78% reduction in the computational cost, compared to the single-machine setup. In future, the work can be extended to efficiently predict the landscape pattern from high-resolution remote sensing imagery.

REFERENCES

[1] B. P. Salmon, J. C. Olivier, K. J. Wessels, W. Kleynhans, F. Van den Bergh, and K. C. Steenkamp, "Unsupervised land cover change detection: Meaningful sequential time series analysis," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 2, pp. 327–335, Jun. 2011.
[2] A. Rahman, S. P. Aggarwal, M. Netzband, and S. Fazal, "Monitoring urban sprawl using remote sensing and GIS techniques of a fast growing urban centre, India," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 1, pp. 56–64, Mar. 2011.
[3] M. C. Hansen and T. R. Loveland, "A review of large area monitoring of land cover change using landsat data," *Remote Sens. Environ.*, vol. 122, pp. 66–74, 2012.
[4] B. Basnet and A. Vodacek, "Tracking land use/land cover dynamics in cloud prone areas using moderate resolution satellite data: A case study in central africa," *Remote Sens.*, vol. 7, no. 6, pp. 6683–6709, 2015.

[5] J. M. Read and N. S.-N. Lam, "Spatial methods for characterising land cover and detecting land-cover changes for the tropics," *Int. J. Remote Sens.*, vol. 23, no. 12, pp. 2457–2474, 2002.
[6] P. Griffiths, S. van der Linden, T. Kuemmerle, and P. Hostert, "A pixel-based landsat compositing algorithm for large area land cover mapping," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 5, pp. 2088–2101, Oct. 2013.
[7] M. Das and S. K. Ghosh, "A cost-efficient approach for measuring Moran's index of spatial autocorrelation in geostationary satellite data," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2016, pp. 5913–5916.
[8] S. Su, Z. Jiang, Q. Zhang, and Y. Zhang, "Transformation of agricultural landscapes under rapid urbanization: A threat to sustainability in Hang-Jia-Hu Region, China," *Appl. Geography*, vol. 31, no. 2, pp. 439–449, 2011.
[9] K. Overmars, G. De Koning, and A. Veldkamp, "Spatial autocorrelation in multi-scale land use models," *Ecol. Model.*, vol. 164, no. 2, pp. 257–270, 2003.
[10] H. Estiri, "Tracking urban sprawl: Applying Moran's I technique in developing sprawl detection models," in *Proc. 43rd Annu. Conf. Envir. Des. Res. Assoc. EDRA*, 2012, pp. 47–53.
[11] J. P. Pierre, C. J. Abolt, and M. H. Young, "Impacts from above-ground activities in the eagle ford shale play on landscapes and hydrologic flows, La Salle County, Texas," *Environ. Manage.*, vol. 55, no. 6, pp. 1262–1275, 2015.
[12] M. Chen, J. C. Rowland, C. J. Wilson, G. L. Altmann, and S. P. Brumby, "Temporal and spatial pattern of Thermokarst lake area changes at Yukon Flats, Alaska," *Hydrol. Processes*, vol. 28, no. 3, pp. 837–852, 2014.
[13] S. A. Roberts, G. B. Hall, and P. H. Calamai, "Analysing forest fragmentation using spatial autocorrelation, graphs and GIS," *Int. J. Geograph. Inf. Sci.*, vol. 14, no. 2, pp. 185–204, 2000.
[14] Y.-H. Tsai, "Quantifying urban form: Compactness versus' sprawl'," *Urban Stud.*, vol. 42, no. 1, pp. 141–161, 2005.
[15] L. Altieri, D. Cocchi, G. Pezzi, E. M. Scott, and M. Ventrucci, "Urban sprawl scatterplots for urban morphological zones data," *Ecol. Indicators*, vol. 36, pp. 315–323, 2014.
[16] B. D. Ripley, *Spatial Statistics*, vol. 575. New York, NY, USA: Wiley, 2005.
[17] Y. Chen, "New approaches for calculating Moran's index of spatial autocorrelation," *PloS One*, vol. 8, no. 7, 2013, Art. no. e68336.
[18] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, 2008.
[19] "USGS EarthExplorer: Land Processes Distributed Active Archive Center," 2014. [Online]. Available: https://lpdaac.usgs.gov/data_access/usgs_earthexplorer. Accessed on: Jun. 9, 2015.
[20] M. Das and S. K. Ghosh, "Deep-STEP: A deep learning approach for spatiotemporal prediction of remote sensing data," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 1984–1988, Dec. 2016.
[21] R, "R-3.2.2 for Windows (32/64 bit)," 2015. [Online]. Available: https://cran.r-project.org/bin/windows/base/old/3.2.2/. Accessed on: Aug. 27, 2015.
[22] ESRI, "ArcGIS for desktop," 2015. [Online]. Available: http://www.esri.com/software/arcgis/ arcgis-for-desktop. Accessed on Aug. 11, 2015.
[23] K. McGarigal, S. Cushman, M. Neel, and E. Ene, *FRAGSTATS: Spatial Pattern Analysis Program for Categorical Maps* (Computer Program), Univ. Massachusetts, Amherst, MA, USA, 2013.

Monidipa Das (S'14) received the M.E. degree in computer science and engineering from the Indian Institute of Engineering Science and Technology, Shibpur, India, in 2013. She is currently working toward the Ph.D. degree with the Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur, India.

Her research interests include spatial and spatiotemporal data mining, soft computing, and machine learning.

Ms. Das is a member of the IEEE Computational Intelligence Society.

Soumya K. Ghosh (M'04) received the Ph.D. and M.Tech. degrees from Department of Computer Science and Engineering, Indian Institute of Technology (IIT), Kharagpur, India. Presently, he is a Professor with Department of Computer Science and Engineering, IIT, Kharagpur. Before joining IIT Kharagpur, he worked for the Indian Space Research Organization in the area of satellite remote sensing and geographic information systems. He has more than 200 research papers in reputed journals and conference proceedings. His research interests include spatial informatics and spatial web services. Dr. Ghosh is a member of IEEE Geoscience and Remote Sensing Society.